

VEB
KOMBINAT ZENTRONIK
RECHENELEKTRONIK
MEININGEN/ZELLA MEHLIS

PROGRAMMIERUNG UND BEDIENUNG DES DIGITALEN KLEINRECHNERS C 8205

HEFT 4

BAUSTEIN-INTERPRETIERSYSTEM

Programmierung und Bedienung

des digitalen Kleinrechenautomaten

C 8205

Heft 4

Bausteininterpretiersystem

Inhaltsverzeichnis

0. Einleitung

Der interne Befehlsschlüssel des C 8205 läßt von den Grundrechenarten nur die Addition und Subtraktion von Festkommazahlen zu. Für einen sinnvollen Einsatz des Rechners in der Praxis ist jedoch Voraussetzung, daß man Festkommazahlen auch multiplizieren und dividieren kann.

Bei der Rechnung mit Festkommazahlen sind die Intervalle der zu verarbeitenden Zahlen stark eingeschränkt. Beim Einsatz des Automaten für Aufgabengebiete, bei denen die Größenordnung der Zahlen sehr stark variiert, wird eine Gleitkommadarstellung benutzt.

Da sich die verdrahtete Ein- und Ausgabe des Rechners nur auf ein Zeichen beziehen, müssen für die Ein- und Ausgabe von ganzen Worten ebenso wie für die Rechenoperationen mit Gleitkommazahlen spezielle Programme aufgestellt werden. Das Bausteininterpretiersystem besteht aus solchen Standardprogrammen für die Ein- und Ausgabe, für die Fest- und Gleitkommarechnung sowie Bedienungs- und Kontrollprogrammen, die für das Eingeben und Testen von Programmen benötigt werden. Gleichartige Programme sind zu sogenannten Bausteinen zusammengefaßt. Diese Bausteine können vom Benutzer entsprechend dem Bedarf zusammengesetzt werden.

Das Bausteininterpretiersystem enthält sowohl Programme für die Ein- und Ausgabe im Fernschreib-Code (5-Kanal-Code) als auch im R 300-Lochband-Code (8-Kanal-Code).

Bei der Ausarbeitung aller Bausteine – speziell bei den Programmen für die Fest- und Gleitkommaoperationen – wurde besonderer Wert auf Zeitoptimalität gelegt.

Alle Bausteine des Systems sind so aufgebaut, daß sie auf sperrbaren Speicherplätzen abgespeichert und gegen Überschreiben gesichert werden können. Die Arbeitszellen der Programme befinden sich auf den letzten vier Spuren des Speichers. Das Protokollprogramm benutzt zusätzlich Speicherplätze der Spur 7500–7537 als Arbeitszellen.

Für den Ansprung der verschiedenen Ein- und Ausgabeprogramme sowie der Rechenprogramme gibt es feste Ansprungzellen, die auf den Zellen 7540–7577 untergebracht sind.

1. Zahlwortinterpretation

Das interne Zahlwort stellt beim C 8205 eine Dualzahl dar, bei der das Komma an einer festen Stelle steht. Die Kommastellung kann im Prinzip durch den Benutzer gewählt werden. Sie ist bei der Programmierung von vornherein zu berücksichtigen. Grundsätzlich kann ein Zahlwort in beliebiger Weise interpretiert werden, auch als Zahl mit beweglichem Komma.

Beim Bausteininterpretiersystem wird das interne Zahlwort als Zahl mit festem Komma, als Gleitkommazahl oder als Zahl in Tetradendarstellung aufgefaßt. Für die interne Darstellung eines Zahlwortes stehen 33 Dualstellen zur Verfügung, die mit Null oder Eins besetzt sein können. Für diese 33 dualen Informationen gibt es eine der jeweiligen Bedeutung des Zahlwortes entsprechende kürzere externe Darstellung, die bei der Programmierung im allgemeinen verwandt wird. Die Umwandlung aus der externen in die interne Darstellung und umgekehrt erfolgt bei der Ein- bzw. Ausgabe durch den Automaten selbst. Bei der Behandlung der verschiedenen Zahlwortinterpretationen wird sowohl auf die interne als auch die externe Darstellung eingegangen.

1.1. Triadendarstellung

1.1.1. Intern

Das Maschinenwort besteht, unabhängig von seiner speziellen Bedeutung, aus 33 Bits, die mit 0 bis 32 von links nach rechts durchnumeriert sind (Abb. 1). Jede dieser 33 Dualstellen ist mit einer der beiden Dualziffern Null oder Eins besetzt. Insgesamt gibt es 2^{33} voneinander verschiedene Maschinenworte.

1.1.2. Extern

Ohne Berücksichtigung der speziellen Bedeutung eines Maschinenwortes kann dieses durch eine externe Darstellung in oktaler Form beschrieben werden. Jeweils drei Dualziffern werden zu einer Oktalziffer (Triade) zusammengefaßt, so daß die den 33 Dualziffern entsprechende externe Darstellung aus 11 Oktalziffern besteht. Die Oktalziffern erhalten die Nummern 0 bis 10. Die Zuordnung der Wortstellen ist aus Abb. 2 ersichtlich. In Abb. 3 sind ein duales Maschinenwort und ein Wort in Triadendarstellung gegenübergestellt.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32

Abb. 1

0 1 2	3 4 5	6 7 8	9 10 11	12 13 14	15 16 17	18 19 20	21 22 23	24 25 26	27 28 29	30 31 32
t_{10}	t_9	t_8	t_7	t_6	t_5	t_4	t_3	t_2	t_1	t_0

Abb. 2

100	111	001	101	000	011	110	010	100	101	011
4	7	1	5	0	3	6	2	4	5	3

Abb. 3

Die angeführte Triadendarstellung ermöglicht die externe Darstellung eines Maschinenwortes, die unabhängig von der Bedeutung desselben ist. Sie ist für jedes interne Wort anwendbar.

1.2. Zahl in der Darstellung Festkomma hinten

1.2.1. Intern

Man spricht von einer Festkommadarstellung, wenn jeder Ziffer einer Zahl ein fester Stellenwert zugeordnet wird. Beim C 8205 werden den in Abschnitt 1.1.1. angegebenen 33 Dualstellen die folgenden Stellenwerte zugeordnet (dual betrachtet):

z_0	z_1	z_2	z_3	z_4	z_5	z_6	z_7	z_8	z_9	z_{10}	z_{11}	z_{12}	z_{13}	z_{14}	z_{15}	z_{16}	z_{17}	z_{18}	z_{19}	z_{20}	z_{21}	z_{22}	z_{23}	z_{24}	z_{25}	z_{26}	z_{27}	z_{28}	z_{29}	z_{30}	z_{31}	z_{32}
2^{31}	2^{30}	2^{29}	2^{28}	2^{27}	2^{26}	2^{25}	2^{24}	2^{23}	2^{22}	2^{21}	2^{20}	2^{19}	2^{18}	2^{17}	2^{16}	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	

Abb. 4

Die Dualstelle z_{32} wird nicht benutzt, so daß die der Stelle z_i zugeordnete Zweierpotenz 2^{31-i} ist. Die Einheit der größten Stelle z_0 ist 2^{31}, die der kleinsten 2^0. Es können Zahlen aus dem Bereich

$$|z| \leqq 2^{31} - 1 = 2\ 147\ 483\ 647$$

intern durch die Maschine verarbeitet werden. (Die intern noch darstellbare Zahl $z = -2^{31}$ (nur z_0 besetzt) führt bei der Dezimalausgabe sowie der Multiplikation und Division zum Überlauf.)
Positive ganze Zahlen können in der Form

$$z = \sum_{i=1}^{31} z_i \cdot 2^{31-i}$$

geschrieben werden. Die kleinste von Null verschiedene positive Zahl ist $2^0 = 1$. Negative Zahlen werden im Zweierkomplement dargestellt. Positive Zahlen sind durch $z_0 = 0$, negative Zahlen durch $z_0 = 1$ gekennzeichnet.

Das Komma steht am Ende des Wortes, d. h. zwischen der letzten durch die Zahl besetzten Dualstelle z_{31} und z_{32}.

1.2.2. Extern

Die zugehörige Externdarstellung erfolgt dezimal. Die Konvertierung ins Dualsystem wird bei der Eingabe durch das Leseprogramm ausgeführt. Die externe Zahl besteht aus dem Vorzeichen und maximal 10 Dezimalziffern:

$$d = (-)\ d_9 d_8 d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0 \cdot$$

Der externe Zahlenbereich ist durch

$$|d| \leqq 2^{31} - 1 = 2\ 147\ 483\ 647$$

beschränkt.

Beispiel: internes Wort: 00...001011010
zugehöriges externes Wort: 45
externes Wort: —45
zugehöriges internes Wort: 11...110100110

Das zu —45 gehörende interne Wort erhält man durch Umrechnung der Zahl 45 ins Dualsystem, stellenweiser Negation, Addition einer Eins in der letzten Stelle und Linksverschiebung um eine Dualstelle.

1.3. Zahl in Gleitkommadarstellung

Jede Zahl kann in einer halblogarithmischen Schreibweise als Produkt aus Mantisse m und Basispotenz B mit dem Exponenten e dargestellt werden:

$$z = (-)\ m \cdot B^e .$$

Dabei wird für m die Normalisierungsvorschrift

$$B^{-1} \leqq m < 1 \qquad (\text{für } z \neq 0) \text{ vereinbart.}$$

Die Mantisse m bestimmt die Ziffernfolge von z und der Exponent e legt fest, an welcher Stelle der Mantisse das Komma steht. B stellt die Basis des Zahlensystems dar.

1.3.1. Intern

Die interne Zahlendarstellung beruht auf der Schreibweise

$$z = (-)\ m \cdot 2^{e}$$

Damit ergibt sich für die Mantisse die Normalisierungsvorschrift (dezimal betrachtet):

$$0{,}5 \leqq m < 1 \qquad (\text{für } z \neq 0) \ .$$

Exponent, Betrag der Mantisse und Vorzeichen werden separat gespeichert.

Der Exponent e umfaßt in der internen Darstellung die Stellen z_1 bis z_6. Für die Mantisse stehen die Stellen z_7 bis z_{32} zur Verfügung.

Das Vorzeichen ist aus der Zahl herausgelöst und wird im Bit z_0 gespeichert.

Die Gleitkommanull entspricht in der internen Darstellung der dualen Null, d. h. alle Dualstellen sind mit Null besetzt.

Für die Vorzeichenstelle gilt:

$$\text{für } z \geqq 0 \text{ ist } z_0 = 0$$
$$\text{für } z < 0 \text{ ist } z_0 = 1 \ .$$

Um Vorzeichenrechnungen im Exponenten zu vermeiden, wird der tatsächliche Exponent e intern um $2^5 = 32$ erhöht angegeben:

$$e' = e + 32 \ .$$

Zwischen den einzelnen Dualstellen und der internen Gleitkommadarstellung ergibt sich somit ein Zusammenhang entsprechend Abb. 5

z_0	z_1	z_2	z_3	z_4	z_5	z_6	z_7	z_8	z_9	...	z_{30}	z_{31}	z_{32}
V	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	...	2^{-24}	2^{-25}	2^{-26}

Vz Exponent e' Mantisse m

Abb. 5

Die Normalisierungsvorschrift $0{,}5 \leqq m < 1$ $(z \neq 0)$ besagt, daß bei allen Gleitkommazahlen mit Ausnahme der Gleitkommanull das Bit z_7 stets mit Eins besetzt sein muß.

Der tatsächliche Exponent e kann das Intervall

$$-32 \leqq e \leqq 31$$

durchlaufen und der Mantissenbereich ist durch

$$0{,}5 \leqq m \leqq 1 - 2^{-26} \qquad (\text{für alle } z \neq 0)$$

beschränkt.

Daraus ergibt sich der interne Zahlenbereich

$$|z| \leqq (1 - 2^{-26}) \cdot 2^{31} = 2^{31} - 2^{5} \ .$$

Die kleinste darstellbare von Null verschiedene positive Zahl ist

$$z = 0{,}5 \cdot 2^{-32} = 2^{-33} \ .$$

1.3.2. Extern

Für die externe Darstellung einer Gleitkommazahl existieren zwei Möglichkeiten. Sie kann als Zahl in üblicher Dezimalschreibweise oder als Zahl in Gleitkommadarstellung angegeben werden, die sich sowohl für die Mantisse als auch den Exponenten auf das Dezimalsystem bezieht:

$$z = (-)\ m \cdot 10^{E}\ .$$

Es gilt die Normalisierungsvorschrift:

$$0{,}1 \leqq m < 1 \qquad (\text{für } z \neq 0)\ .$$

Der Exponent E wird nicht erhöht. Er besteht aus dem Vorzeichen und dem Betrag von E.

$$E = (-)\ e,\ e = |E|$$

Die externe Gleitkommazahl setzt sich zusammen aus dem Vorzeichen der Mantisse, maximal acht dezimalen Mantissenziffern, dem Exponentenvorzeichen und dem Betrag des Exponenten E.

$$z = (-)\ m_1\, m_2\, m_3\, m_4\, m_5\, m_6\, m_7\, m_8\ (-)\ e$$

Da der Exponent E bei der Eingabe nicht größer als 8 sein darf, ist der Eingabezahlenbereich durch

$$|z| \leqq (1 - 10^{-8}) \cdot 10^{8}$$

beschränkt.

Die kleinste positive Zahl $\neq 0$, die eingegeben werden kann, ist

$$z = 0{,}1 \cdot 10^{-8} = 10^{-9}.$$

Bei der externen Darstellung in Dezimalschreibweise wird die Gleitkommazahl in der Form

$$z = (-)\ a,\ b$$

angegeben.

Der Eingabezahlenbereich ist wie bei der Gleitkommadarstellung durch

$$|z| \leqq (1 - 10^{-8}) \cdot 10^{8}$$

beschränkt. Die größte eingebbare Zahl ist 99 999 999, die kleinste positive Zahl $\neq 0$ ist 0,000000001.

1.4. Zahl in Tetradendarstellung

Bei der Tetradendarstellung von Dezimalzahlen wird jede Dezimalziffer in Form einer vierstelligen Dualzahl (Tetrade) geschrieben. Sie unterscheidet sich von der reinen Dezimaldarstellung dadurch, daß für die Beschreibung einer Ziffer nicht ein, sondern vier Zeichen verwendet werden. Bei der internen Darstellung können in den 33 Dualstellen des Maschinenwortes 8 Tetraden untergebracht werden. Die Zuordnung zwischen den in Abb. 1 angegebenen Wortstellen und den einzelnen Tetraden sieht folgendermaßen aus:

$Z_0\ Z_1\ Z_2\ Z_3$	$Z_4\ Z_5\ Z_6\ Z_7$	$Z_8\ Z_9\ Z_{10}\ Z_{11}$	$Z_{12}\ Z_{13}\ Z_{14}\ Z_{15}$	$Z_{16}\ Z_{17}\ Z_{18}\ Z_{19}$	$Z_{20}\ Z_{21}\ Z_{22}\ Z_{23}$	$Z_{24}\ Z_{25}\ Z_{26}\ Z_{27}$	$Z_{28}\ Z_{29}\ Z_{30}\ Z_{31}$	Z_{32}
T_7	T_6	T_5	T_4	T_3	T_2	T_1	T_0	

Abb. 6

Die Dualstelle z_{32} wird wie bei der internen Festkommazahl nicht besetzt.

Bei der externen Darstellung besteht die Zahl aus maximal acht Dezimalziffern ohne Vorzeichen.

Der Zahlenbereich ist durch

$$0 \leq z \leq 10^8 - 1$$

beschränkt.

Beispiel:

externes Wort	2	8	1	3	6	9	0	5	
internes Wort	0010	1000	0001	0011	0110	1001	0000	0101	0

Im Bausteininterpretiersystem sind nur Programme für die Ein- und Ausgabe von Tetraden enthalten.

Rechenoperationen mit Zahlen in Tetradendarstellung können nicht durchgeführt werden.

2. Eingabe

Da beim C 8205 nur die Eingabe eines Zeichens fest verdrahtet ist, sind spezielle Leseprogramme erforderlich, um ganze Worte – z. B. Befehle oder Zahlen – eingeben zu können.

Das Bausteininterpretiersystem enthält Programme für die Eingabe im 5-Kanal-Fernschreib-Code (Internationales Telegraphenalphabet Nr. 2) und für die Eingabe im R 300-Code (8-Kanal-Code).

Die Eingabe kann über die Tastatur, die Schreibmaschine oder die beiden Lochbandleser erfolgen.

Die beim BIS verwendete Eingabeorganisation ermöglicht mit Hilfe von Eingabekennzeichen die Eingabe von Worten mit unterschiedlicher Bedeutung (Befehle, Zahlen in verschiedenartiger Darstellung). Außerdem sind im BIS Unterprogramme enthalten, die gestatten, Zahlen eines bestimmten Typs ohne die Angabe eines Kennzeichens einzulesen.

Die einzugebenden Informationen werden im allgemeinen durch eine Wortmarke „WM" abgeschlossen.

In der 5-Kanal-Version wird als Wortmarke das Zeichen Punkt „." benutzt.

In der 8-Kanal-Version wird bei Maschinenbefehlen, Externbefehlen und Adressen das Zeichen Punkt (.) verwendet, während bei Zahlworten eines der beiden Zeichen Punkt (.) oder Tabulator (Tab) zur Markierung des Wortendes benutzt werden kann.

Bei den Eingaben (Adressen, Befehlen und Zahlen) können i. a. führende Nullen im Wort weggelassen werden. Eine Ausnahme stellen die im Abschnitt 5. beschriebenen Externbefehle dar. Für die Eingabe von Zahlen in allen Darstellungen gilt, daß ein positives Vorzeichen nicht eingegeben werden darf, während ein negatives Vorzeichen an einer beliebigen Stelle auch innerhalb der Zahl vor der Wortmarke stehen kann.

2.1. Eingabe mit Kennzeichen

Um Befehle und Zahlen unterschiedlicher Darstellung eingeben zu können, muß es dem Automaten möglich sein, die ihm mitgeteilten Informationen zu unterscheiden. Außerdem müssen die eingelesenen Worte abgespeichert werden.

Zu diesem Zweck wurden für die Eingabe besondere Eingabekennzeichen (K_1 und K_2) und Hilfsadressen (Relativ-, Hauptleit-, Leitadressen) eingeführt.

2.1.1. Relative Programmierung

Fast alle Programme werden relativ programmiert. Dadurch ist es möglich, ein aufgestelltes Programm auf einen beliebigen Speicherbereich einzugeben.

Bei der relativen Programmierung geht man bei der Numerierung eines Programms von der Adresse 0000 aus, und Befehle, deren Adresse sich auf diese Anfangsadresse beziehen, werden markiert. Bei der Eingabe eines Programms wird der Adreßteil jedes so markierten Befehls um die Adresse erhöht, ab der die Abspeicherung beginnt. Diese Adresse wird beim C 8205 als Hauptleit- oder 1. Leitadresse bezeichnet.

Die Hauptleitadresse erfüllt zwei Funktionen. Sie bestimmt die Adresse, ab der das Programm gespeichert werden soll, und sie dient zur Adressenänderung 1. Art.

Für die Adressenänderung 1. Art stehen beim BIS außerdem noch weitere 8 Adressen – sogenannte Leitadressen – zur Verfügung.

Diese Adressen verwendet man z. B. als Anfangsadressen für Zahlenfelder, die vom Programm benötigt werden, und zur Parameterversorgung von Unterprogrammen.

Im Gegensatz zur Hauptleitadresse werden die acht Leitadressen ausschließlich für die Adressenänderung 1. Art benutzt.

2.1.2. Eingabekennzeichen und deren Zuordnung

Zu jedem Befehl und jeder Zahl, die in den Automaten eingegeben werden sollen, gehört im allgemeinen ein „Erstes Kennzeichen K_1". Entsprechend diesem Kennzeichen K_1 wird das eingelesene Wort interpretiert und weiterverarbeitet.

Die Eingabekennzeichen werden nicht mit abgespeichert. Sie dienen nur zur Steuerung.

Als K_1 sind folgende Zeichen zugelassen:

Ziffern: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Buchstaben: B, D

Zeichen: Plus (+), Zwischenraum (ZR), Schrägstrich (/), Doppelpunkt (:), Blockmarke (BM)
((BM) tritt nur im R 300-Lochband-Code auf).

Befehle haben für die Adressenänderung 1. Art ein Kennzeichen, das als „Zweites Kennzeichen K_2" bezeichnet wird. K_2 kann die Werte 0 bis 9 annehmen.

Alle definierten Kennzeichen K_1 und K_2 haben bei der Eingabe im Fernschreib-Code und im R 300-Code die gleiche Bedeutung.

$K_1 = 0$ **Eingabe eines Maschinenwortes zur laufenden Speicherung**

Nach der Eingabe des ersten und zweiten Kennzeichens können bis zu 11 Oktalziffern eingegeben werden:

$0K_2 \; t_{10} t_9 t_8 t_7 t_6 t_5 t_4 t_3 t_2 t_1 t_0$.

Nach Eingabe der letzten Oktalziffer muß das Schlußzeichen Punkt „." folgen. Bei der Eingabe kann mit der ersten von Null verschiedenen Ziffer begonnen werden.

Für die Eingabe der Oktalzahl 0 wurde eine Sonderregelung getroffen. Nach Eingabe des ersten Kennzeichens $K_1 = 0$ kann sofort die Wortmarke „Punkt" gegeben werden:

0. (Null-Punkt).

Die eingelesenen Oktalziffern werden ohne Konvertierung abgespeichert und, falls sie einen Internbefehl darstellen, erst später abgearbeitet. Die Unterscheidung, ob das Wort als Befehl oder Zahl aufzufassen ist, hängt von seiner Verwendung ab. Es ist z. B. möglich, daß in einem Programm eine Oktalzahl als Befehl und als Konstante benutzt wird. In Abhängigkeit vom zweiten Kennzeichen K_2 kann vor der Abspeicherung eine Adressenänderung durchgeführt werden. Diese Änderung bezieht sich auf die Oktalstellen t_3 bis t_{10}, die damit den Adreßteil eines Maschinenwortes darstellen.

Das zweite Kennzeichen hat im einzelnen folgende Wirkung:

$K_2 =$ „0" Das eingelesene Wort wird ohne Änderung der Oktalstellen t_3 bis t_{10} abgespeichert. Falls das Wort einen Befehl darstellt, ist die angegebene Adresse absolut.

$K_2 =$ „1" Zu den Oktalstellen t_3 bis t_{10} wird vor der Abspeicherung die Hauptleitadresse (= 1. Leitadresse) addiert (vgl. $K_1 = 2$, $K_2 = 1$).

$K_2 = „2“$ Zu den Oktalstellen t_3 bis t_{10} wird vor der Abspeicherung die 2. Leitadresse addiert (vgl. $K_1 = 2$, $K_2 = 2$).

.
.
.

$K_2 = „9“$ Zu den Oktalstellen t_3 bis t_{10} wird vor der Abspeicherung die 9. Leitadresse addiert (vgl. $K_1 = 2$, $K_2 = 9$).

$K_1 = 1$ **Maschinenbefehl zur sofortigen Abarbeitung**

Nach der Angabe des ersten und zweiten Kennzeichens könnten bis zu 11 Oktalziffern eingegeben werden. Da das eingelesene Wort jedoch als Befehl interpretiert wird, ist nur die Eingabe von maximal 7 Ziffern sinnvoll.

$1\ K_2 \quad t_6\, t_5\, t_4\, t_3\, t_2\, t_1\, t_0\, .$

Wie bei $K_1 = 0$ kann auch hier mit der ersten Oktalziffer ungleich Null begonnen werden, und nach Eingabe der letzten Ziffer muß die Wortmarke Punkt (.) gegeben werden.

Das zweite Kennzeichen K_2 hat dieselbe Bedeutung wie bei $K_1 = 0$.

Ein Sofortbefehl wird nach seiner Eingabe in einer festen Zelle gespeichert und anschließend sofort abgearbeitet. Sofort-Sprungbefehle werden z. B. zum Anspringen von Programmen benötigt.

Wenn der Sofortbefehl kein Sprungbefehl ist, geht der Automat nach Ausführung des Befehls in den Zustand „unbedingter Stopp“. Bei Betätigung der „Starttaste“ (siehe Bedienungsanleitung) wird die Adresse des Sofortbefehls um eine Einheit erhöht, und der dadurch entstandene Befehl wird sofort ausgeführt. Anschließend geht der Automat wieder in den Zustand „unbedingter Stopp“. Mit Hilfe der Tastenkombination „HGR“ kann zur Eingabe eines neuen ersten Kennzeichens zurückgekehrt werden.

So kann z. B. durch einen Sofort-Lesebefehl der Inhalt des durch die Adresse angegebenen Speicherplatzes in den Akkumulator gebracht und über ein Hilfsprogramm ausgegeben werden.

$K_1 = 2$ **Hilfsadressen**

Bei der Eingabe von Hilfsadressen können nach dem ersten und zweiten Kennzeichen maximal 8 Oktalziffern angegeben werden, die zwei oktale Adressen darstellen:

$2\ K_2 \quad b_1\, b_2\, b_3\, b_4 \quad a_1\, a_2\, a_3\, a_4\, .$

Führende Nullen brauchen nicht eingegeben zu werden.

Als Wortmarke wird das Zeichen Punkt „.“ benutzt. Die 8 Oktalziffern werden auf festen Speicherplätzen in den Oktalstellen t_3 bis t_{10} gespeichert.

Die Möglichkeit, Hilfsadressen mit 8 Ziffern eingeben zu können, bietet den Vorteil, daß auch Änderungen mit den ersten vier Stellen $t_{10}\, t_9\, t_8\, t_7$ einer Oktalzahl durchgeführt werden können (vgl. $K_1 = 0$).

Für das Kennzeichen K_2 sind die Ziffern 0 bis 9 zugelassen. Sie sind im einzelnen den folgenden Hilfsadressen zugeordnet:

K_2 = „0" Relativadresse

Bei der Relativadresse ist nur die Angabe der vier Oktalziffern $a_1\,a_2\,a_3\,a_4$ sinnvoll. Das Eingabebild hat die Gestalt:

20 $a_1\,a_2\,a_3\,a_4$.

Mit Hilfe der Relativadresse kann die Abspeicherung in natürlicher Reihenfolge unterbrochen werden. Sie bewirkt einen Neuaufbau des Abspeicherbefehls (vgl. $K_1 = 2$, $K_2 = 1$). Die Abspeicherung wird ab der Adresse fortgesetzt, die sich aus der Summe von Hauptleit- und Relativadresse ergibt. Die Hauptleitadresse wird dabei nicht verändert, so daß sie weiterhin verwendet werden kann. Die Relativadresse steht danach nicht mehr zur Verfügung.

K_2 = „1" Hauptleitadresse (1. Leitadresse)

Die Hauptleitadresse kann mit maximal 8 Ziffern eingegeben werden.

21 $b_1\,b_2\,b_3\,b_4$ $a_1\,a_2\,a_3\,a_4$.

Mittels der Hauptleitadresse wird der Abspeicherbefehl für die einzugebenden Worte aufgebaut. Die Speicherung beginnt bei der durch die Adresse $a_1\,a_2\,a_3\,a_4$ angegebenen Zelle. Die Adresse $b_1\,b_2\,b_3\,b_4$ hat keinen Einfluß auf die Abspeicherung eines Wortes. Nach jeder erfolgten Eingabe eines zu speichernden Wortes wird die Adresse des Abspeicherbefehls um eine Einheit erhöht, so daß eine Abspeicherung in natürlicher Reihenfolge gewährleistet ist. Die Speicherung in natürlicher Reihenfolge wird unterbrochen durch die Eingabe einer Relativadresse oder einer neuen Hauptleitadresse. Die Hauptleitadresse dient außerdem zur Adressenänderung 1. Art.

Durch die Angabe einer 8stelligen Hauptleitadresse sind bei der Eingabe von Maschinenworten nicht nur Adressenänderungen mit der eigentlichen Adresse eines Befehlswortes ($t_6\,t_5\,t_4\,t_3$) möglich, sondern auch mit den Oktalstellen $t_{10}\,t_9\,t_8\,t_7$ (vgl. $K_1 = 0$).

Die Hauptleitadresse und der Speicherbefehl werden in festen Zellen gespeichert. Die Hauptleitadresse wird im allgemeinen erst durch die Eingabe einer neuen Hauptleitadresse zerstört.

K_2 = „2" 2. Leitadresse

K_2 = „3" 3. Leitadresse

.
.
.

K_2 = „9" 9. Leitadresse

Die zweite bis neunte Leitadresse können wie die Hauptleitadresse maximal 8 Oktalziffern umfassen. Das Eingabebild hat die Gestalt:

$2K_2$ $b_1\,b_2\,b_3\,b_4$ $a_1\,a_2\,a_3\,a_4$.

Die zweite bis neunte Leitadresse haben keinen Einfluß auf die laufende Speicherung von Worten. Sie werden ausschließlich zur Adressenänderung verwendet.

$K_1 = 3$ **Dezimalzahl (intern Konvertierung in eine Gleitkommazahl)**

Mit Hilfe des Kennzeichens $K_1 = 3$ kann eine Zahl in Dezimalschreibweise eingelesen werden. Das Eingabebild hat die Form:

3 (—) a, b WM bzw. 3 (—) a WM

Die Zahl wird intern in eine Gleitkommazahl umgewandelt.

Der Eingabezahlenbereich ist durch

$|z| \leqq (1 - 10^{-8}) \cdot 10^8$

beschränkt (vgl. Abschnitt 1.3.2.).

Nach der ersten von Null verschiedenen Ziffer dürfen höchstens noch 7 Ziffern stehen.

Bei Zahlen mit $|z| < 1$ können zwischen Komma und erster von Null verschiedener Ziffer bis zu acht Nullen auftreten.

Die Zahl Null wird mit

3 0 WM

eingegeben.

$K_1 = 4$ **Zahl in Gleitkommadarstellung**

Eine Gleitkommazahl umfaßt maximal 14 Zeichen:

— erstes Kennzeichen
— negatives Vorzeichen der Mantisse
— maximal acht dezimale Mantissenziffern
— Mantissenschlußzeichen
— negatives Vorzeichen des Exponenten
— Betrag des Exponenten E
— Exponentenschlußzeichen

4 (—) $d_1\ d_2\ d_3\ d_4\ d_5\ d_6\ d_7\ d_8$ WM (—) e WM

Ein negatives Vorzeichen des Exponenten muß vor diesem eingegeben werden.

Alle Gleitkommazahlen mit Ausnahme der Null sind normalisiert einzugeben, d. h., in jedem Fall muß $d_1 \neq 0$ sein. Das Mantissenschlußzeichen kann nach der letzten Ziffer ungleich Null gegeben werden.

Die Null ist als Gleitkommazahl in der Form

4 0 WM

einzugeben.

Der Eingabezahlenbereich ist wie bei $K_1 = 3$ durch

$|z| \leqq (1 - 10^{-8}) \cdot 10^8$

beschränkt.

$K_1 = 5$ **Dezimalzahl (intern Konvertierung in eine Festkommazahl, K. h.)**

Es können ganze Zahlen aus dem Bereich

$|d| \leqq 2^{31} - 1 = 2\ 147\ 483\ 647$

eingegeben werden (vgl. Abschnitt 1.2.2.).

Diese bestehen aus maximal 13 Zeichen:

— erstes Kennzeichen
— negatives Vorzeichen
— maximal 10 Dezimalziffern
— Wortmarke

5 (—) $d_9\ d_8\ d_7\ d_6\ d_5\ d_4\ d_3\ d_2\ d_1\ d_0$ WM

Tritt ein Komma auf, so wird dieses überlesen. Die eingelesene Zahl wird in jedem Fall als ganzzahlig aufgefaßt.

Bemerkung: Die Kommastellung der einzugebenden Zahl ist vom Programmierer bzw. bei der Erstellung der Daten zu berücksichtigen.

$K_1 = 6$ **Dezimalzahl (intern Konvertierung in eine Zahl in Tetradendarstellung)**

Es können ganze positive Zahlen aus dem Bereich

$d \leqq 10^8 - 1$

eingelesen werden.

Das Eingabebild umfaßt maximal 10 Zeichen:
- erstes Kennzeichen
- maximal 8 Dezimalziffern
- Wortmarke

6 $d_7\,d_6\,d_5\,d_4\,d_3\,d_2\,d_1\,d_0$ WM

Ein auftretendes Komma wird überlesen. Jede Dezimalziffer wird durch eine Tetrade dargestellt und in dieser Form abgespeichert.

$K_1 = 7$ **Dezimalzahl (intern Konvertierung in eine Festkommazahl, K. h.)**

Der Eingabezahlenbereich ist durch

$|d| \leqq 10^8 - 1$

beschränkt. Eine Zahl besteht aus maximal 11 Zeichen:
- erstes Kennzeichen
- negatives Vorzeichen
- maximal 8 Dezimalziffern
- Wortmarke

7 (—) $d_7\,d_6\,d_5\,d_4\,d_3\,d_2\,d_1\,d_0$ WM

Wie bei $K_1 = 5$ und $K_1 = 6$ wird auch bei $K_1 = 7$ ein Komma überlesen und die Zahl als ganze Zahl angesehen. Nach dem Einlesen und Darstellung jeder Dezimalziffer durch eine Tetrade wird die Zahl in eine Festkommazahl, Komma hinten konvertiert.

Mit Hilfe von $K_1 = 7$ ist eine schnellere Eingabe von Festkommazahlen, Komma hinten, möglich als mit $K_1 = 5$.

$K_1 = 8$ **Dezimalzahlen ohne Kennzeichen im Satz (intern Konvertierung in Gleitkommazahlen)**

Nach Eingabe dieses Kennzeichens $K_1 = 8$ können hintereinander Dezimalzahlen ohne erstes Kennzeichen $K_1 = 3$ eingelesen werden. Die Zahlen werden intern in Gleitkommazahlen konvertiert (vgl. $K_1 = 3$).

$K_1 = 9$ **Dezimalzahlen ohne Kennzeichen im Satz (intern Konvertierung in Festkommazahlen, K. h.)**

Nach $K_1 = 9$ können hintereinander Dezimalzahlen ohne erstes Kennzeichen $K_1 = 5$ eingelesen werden (Sie werden intern in Festkommazahlen, Komma hinten, konvertiert (vgl. $K_1 = 5$).

Bei $K_1 = 8$ und $K_1 = 9$ können nur jeweils Zahlen gleichen Typs eingegeben werden. Die Abspeicherung erfolgt entsprechend der Hauptleitadresse, die vor dem ersten Kennzeichen $K_1 = 8$ oder $K_1 = 9$ anzugeben ist.

Bei der Eingabe im Fernschreib-Code wird ein Satz durch das Zeichen „Doppelpunkt (:)" abgeschlossen, während bei Verwendung des R 300-Codes die Satzeingabe durch die Zeichen „Satzmarke (SM)", „Blockmarke (BM)" und „Doppelpunkt (:)" beendet werden kann. Nach der Eingabe eines dieser Zeichen ist der Automat bereit, ein neues erstes Kennzeichen K_1 einzulesen.

$K_1 = +$ **Externbefehl**

Mit Hilfe des ersten Kennzeichens $K_1 = +$ werden beim Bausteininterpretiersystem Externbefehle eingegeben. Diese können maximal 7 Zeichen umfassen:

- erstes Kennzeichen +
- zwei oktale Ziffern O_1 O_2
- maximal 3 Dezimalziffern p_1 p_2 p_3
- Wortmarke Punkt (.)

$+ \; O_1 O_2 \quad p_3 p_2 p_1 .$

Die beiden Oktalziffern O_1 und O_2 müssen in jedem Fall angegeben werden. Das heißt, auch bei $O_1 = 0$ ist O_1 einzugeben. Bei der Eingabe der Parameter p_1, p_2 und p_3 können führende Nullen weggelassen werden. Wird z. B. die Angabe des Parameters p_1 gefordert, während p_2 und p_3 nicht benötigt werden, so hat das Eingabebild die Gestalt:

$+ \; O_1 O_2 \quad p_1 .$

Der Externbefehl stellt einen Pseudobefehl dar. Bei der Eingabe wird der zugehörige Internbefehl aus einer Tabelle ausgesucht und nach Addition der Parameter p_1, p_2 und p_3 abgespeichert.

Wegen ihrer Bedeutung werden die Externbefehle in einem besonderen Abschnitt behandelt (vgl. 5.).

K_1 = B, K_1 = D **Hilfsprogramme**

Mit Hilfe dieser Kennzeichen werden Bedienungs- und Kontrollprogramme angesprungen. Ein zweites Kennzeichen bewirkt die Unterscheidung zwischen den einzelnen Programmen.

(Die Beschreibung der Programme erfolgt im Abschnitt 6.).

K_1 = ZR **Löschen**

Das erste Kennzeichen K_1 = ZR wird zur Gliederung des Lochbandes benutzt. Bei Eingabe von „ZR" wird der Inhalt des Akkumulators gelöscht und zu einer neuen Eingabe übergegangen. Die Verwendung des Zeichens „ZR" wird im Abschnitt 8. beschrieben.

K_1 = / **Eingabe-Ende**

Dieses Zeichen steht im allgemeinen am Ende eines Lochbandes. Bei eingerastetem Schalter „BS" geht der Automat in bedingten Stopp. Durch Lösen von „BS" kann die Eingabe fortgesetzt werden.

2.1.3. Minimalprogramm

Das Minimalprogramm (MP) bildet den Grundbaustein des BIS. Es belegt die Speicherplätze 0–337.

Zum Minimalprogramm gehören neben anderen Programmen das „Kleine Eingabeprogramm" und ein Programm für die Eingabe mit Kennzeichen.

Von den als erste Kennzeichen definierten Zeichen verarbeitet das Minimalprogramm die Kennzeichen K_1 = 0, 1, 2 sowie K_1 = „ZR" und K_1 = „/", d. h. im Minimalprogramm sind nur Leseprogramme für abzuspeichernde und für sofort auszuführende Maschinenworte, für Hilfsadressen sowie für die Zeichen „ZR" und „Schrägstrich (/)" enthalten. Für alle übrigen zugelassenen Kennzeichen sind im Minimalprogramm lediglich Anschlußbedingungen für die diesen Kennzeichen entsprechenden Eingabeprogramme vorhanden.

Wenn z. B. eine Dezimalzahl mit dem ersten Kennzeichen $K_1 = 3$ eingegeben werden soll, so muß vorher das Leseprogramm für Dezimalzahlen abgespeichert werden.

Bei Eingabe eines nicht als K_1 definierten Zeichens als erstes Kennzeichen wird zur Eingabe eines neuen K_1 übergegangen.

2.1.4. Kleines Eingabeprogramm

Um bei einem Wechsel des Interpretationssystems oder beim Übergang vom 5- zum 8-Kanal-Code die meist recht umfangreichen Eingabeprogramme, wie z. B. die Minimalprogramme für die beiden Codes, eingeben zu können, gibt es ein einfaches Leseprogramm. Dieses Programm bezeichnet man als „Kleines Eingabeprogramm (KEP)".

Das Kleine Eingabeprogramm steht auf der Bahn 0 und kann auf dem Speicherplatz 0002 angesprungen werden. Gegen Überschreiben ist es grundsätzlich gesichert (vgl. Bedienungsanleitung). Die Eingabe dieses Programms ist bei vollständig gelöschtem Speicher nur über eine Hilfstastatur möglich.

Das Kleine Eingabeprogramm kann nur die fünf Ziffern und Zeichen „0", „1", Punkt „.", Doppelpunkt „:" und Zwischenraum „ZR" verarbeiten. Es gibt kein Eingabekennzeichen.

Eine Adressenänderung ist nicht möglich. Alle Wörter müssen entsprechend ihrer internen dualen Darstellung durch Kombinationen von „0" und „1" eingegeben werden. Dabei kann mit dem ersten von Null verschiedenen Bit begonnen weden. Das Wort wird im Akkumulator aufgebaut. Als Schlußzeichen werden die beiden Zeichen Punkt und Doppelpunkt benutzt, während das Zeichen Zwischenraum zur Korrektur verwendet wird.

Durch das Schlußzeichen „:" wird die im Akkumulator aufgebaute Information als Befehl auf einem festen Speicherplatz gespeichert (Aufbau des Leitbefehles) und anschließend die Eingabe fortgesetzt. Durch das Schlußzeichen „." wird der Leitbefehl ausgeführt. Falls er kein Sprungbefehl ist, wird anschließend seine Adresse um eine Einheit erhöht und zur Eingabe eines neuen Zeichens übergegangen.

Handelt es sich z. B. bei dem Leitbefehl um einen Transportbefehl $a_1 a_2 a_3 a_4$ 240, so wird durch das Schlußzeichen Punkt das vorher im Akkumulator aufgebaute Wort nach der Zelle $a_1 a_2 a_3 a_4$ abgespeichert. Die Erhöhung des Leitbefehles um eine Adresseneinheit gewährleistet die Abspeicherung in natürlicher Reihenfolge. Diese kann nur durch die Eingabe eines neuen Leitbefehls unterbrochen werden.

Soll der im Akkumulator aufgebaute Befehl sofort abgearbeitet werden, müssen als Schlußzeichen „:" und „." gegeben werden.

Durch das Zeichen „ZR" können Tipp- und Lochfehler korrigiert werden, solange noch kein Schlußzeichen gegeben ist. Dabei wird der Akkumulatorinhalt gelöscht und das Wort kann nochmals eingegeben werden.

2.2. Eingabeunterprogramme

Bei der Eingabe mit Unterprogrammen wird von einem Hauptprogramm durch den Befehl „Sprung mit Rückkehrabsicht" zu einem Leseprogramm gesprungen, das als Unterprogramm aufgebaut ist. Nach dem Einlesen der Information und eines die Eingabe beendenden Schlußzeichens wird die Abarbeitung des Hauptprogrammes fortgesetzt. Die Eingabe kann über die Tastatur, das Schreibwerk oder die beiden Lochbandleser erfolgen.

2.2.1. Eingabeunterprogramme des Minimalprogramms

Im Minimalprogramm sind drei Eingabeprogramme enthalten, die als Unterprogramme verwendet werden können:

	Ansprungzelle
Adresseneingabe	316
Eingabe eines Wortes mit Kennzeichen	262
Eingabe eines Programmstückes	263

Nach dem Anspung des Unterprogramms „Adresseneingabe" ist der Automat bereit, eine maximal achtstellige oktale Adreßkonstante einzulesen. Dabei kann mit der ersten Ziffer ungleich Null begonnen werden. Als Schlußzeichen muß das Zeichen Punkt „." eingegeben werden. Bei einer Adreßkonstante Null ist nur die Eingabe des Schlußzeichens notwendig. Die Konstante steht nach dem Einlesen im Akkumulator und belegt die Wortstellen z_0 bis z_{23}.

Bei Verwendung dieses Eingabeunterprogrammes darf das Zeichen „Zwischenraum" nicht gegeben werden, da „ZR" einen Sprung zum Programm „Kennzeicheneingabe" bewirkt. In diesem Fall gibt es bei Angabe einer falschen Adresse keine Korrekturmöglichkeit.

Bei dem Unterprogramm „Eingabe eines Wortes mit Kennzeichen" erfolgt die Interpretation der einzugebenden Informationen entsprechend den im Abschnitt 2.1.2. beschriebenen Eingabekennzeichen. Es wird nur ein Wort in den Akkumulator eingelesen. Die Abspeicherung oder Weiterverarbeitung muß durch das Hauptprogramm erfolgen. Bei Eingabe einer Relativ- oder Hauptleitadresse mit 20 bzw. 21 wird nicht zum Hauptprogramm zurückgekehrt. Der Automat wartet dann auf die Eingabe eines neuen ersten Kennzeichens. Wird eine Leitadresse (2. bis 9.) eingegeben, so liest der Rechner anschließend ein weiteres Wort mit Kennzeichen ein, ehe zum Hauptprogramm zurückgekehrt wird.

Bei Verwendung des Unterprogramms „Eingabe eines Programmstückes" wird die Regie dem im Minimalprogramm enthaltenen Programm für die Kennzeicheneingabe übergeben. Es können ganze Programme oder auch Datenblöcke eingegeben werden. Die zugehörige Eingabeorganisation wurde im Abschnitt 2.1. erläutert. Die Abspeicherung des Programmstückes erfolgt ab der durch die Hauptleitadresse angegebenen Speicherzelle. Die Hauptleitadresse kann über die Tastatur eingegeben werden, oder sie kann auf dem Lochband stehen.

Die Rückkehr ins Hauptprogramm ist durch Eingabe des Schlußzeichens möglich. Als Schlußzeichen werden im R 300-Code die Zeichen Blockmarke „BM" und Doppelpunkt „:" und im Fernschreib-Code das Zeichen Doppelpunkt verwendet.

Will man an eine andere Stelle des Hauptprogrammes zurückkehren, so kann als Schlußzeichen ein entsprechender Sofortbefehl gegeben werden.

2.2.2. Eingabe einer Zahl ohne Kennzeichen

Um bei der Benutzung der Eingabeunterprogramme sowie aller anderen nicht im Minimalprogramm enthaltenen Programme des BIS dem Programmierer die Arbeit zu erleichtern, sind für den Ansprung der Unterprogramme bestimmte Speicherzellen festgelegt worden. Diese Speicherzellen befinden sich auf der Spur 7540. Durch die Benutzung von festen Ansprungzellen wird umgangen, daß sich der Programmierer bei der Eingabe eines Programmes auf verschiedene Speicherplätze auch die sich ändernden Ansprungadressen zu merken braucht. Außerdem ermöglichen diese festen Ansprungzellen die Verwendung von Externbefehlen.

Für das Einlesen einer Zahl ohne Kennzeichen existieren 6 verschiedene Eingabemöglichkeiten:

		Ansprungzelle
(a)	Eingabe einer Zahl in Gleitkommadarstellung (vgl. $K_1 = 4$)	7575
(b)	Eingabe einer Zahl in Dezimalschreibweise (intern Umwandlung in eine Gleitkommazahl; vgl. $K_1 = 3$)	7574
(c)	Eingabe einer Zahl in Dezimalschreibweise (intern Umwandlung in eine Festkommazahl, K. h.; vgl. $K_1 = 5$)	7573
(d)	Eingabe einer Zahl in Dezimalschreibweise mit Kommainformation (intern Umwandlung in eine Festkommazahl, K. h.)	7572
(e)	Eingabe einer Zahl in Dezimalschreibweise mit maximal 8 Ziffern (intern Umwandlung in eine Festkommazahl, K. h.; vgl. $K_1 = 7$)	7571
(f)	Eingabe einer maximal 8stelligen Zahl in Dezimalschreibweise ohne Vorzeichen (intern Umwandlung in eine Zahl in Tetradendarstellung; vgl. $K_1 = 6$)	7570

Bei Verwendung dieser Eingabeprogramme steht die Zahl nach dem Einlesen im Akkumulator. Die Weiterverarbeitung muß durch das Hauptprogramm erfolgen.

Die unter (d) aufgeführte Eingabemöglichkeit gibt es bei der Eingabe mit Kennzeichen nicht. Bei der Eingabe einer Zahl mittels dieses Leseprogramms ist eine Kommainformation notwendig. Es muß die Anzahl der Stellen nach dem Komma angegeben werden, mit welcher die Zahl eingelesen werden soll. Dieser Parameter p_1 (1 $\leqq p_1$ 15)$\leqq$ steht in den Bits $z_8\, z_9\, z_{10}\, z_{11}$ des Befehls „Sprung mit Rückkehrabsicht nach 7572". (Die Bits z_0 bis z_{11} sind bei einem Maschinenbefehl im allgemeinen nicht besetzt.) Der Parameter p_1 bewirkt, daß die Zahl beim Einlesen mit 10^{p_1} multipliziert wird, auch wenn ihre Stellenzahl hinter dem Komma kleiner als p_1 ist. Ist die Anzahl der Stellen hinter dem Komma größer als p_1 angibt, so hat p_1 keine Wirkung.

Die Zahl wird in eine Festkommazahl, Komma hinten, konvertiert.

Für den Eingabezahlenbereich und die einzugebenden Zeichen gilt das bei $K_1 = 5$ angegebene mit den Ausnahmen, daß kein erstes Kennzeichen $K_1 = 5$ auftreten darf und daß ein Komma nicht überlesen wird.

Beispiel für die Wirkung der Kommainformation:

Als Kommainformation ist $p_1 = 5$ angegeben, d. h. die Zahl soll mit fünf Stellen nach dem Komma eingelesen werden. Der Eingabebefehl hat die Form 00 5 7572 025. . Die einzugebende Zahl ist 31,74, d. h. sie besitzt nur zwei Stellen nach dem Komma. Während der Eingabe wird die Zahl mit $10^{p_1} = 10^5$ multipliziert, so daß danach die Zahl

3174000

im Akkumulator steht.

Ist $p_1 = 1$ angegeben, so steht nach dem Einlesen die Zahl 3174 im Akkumulator.

2.2.3. Satzeingabe von Zahlen ohne Kennzeichen

Auch die Unterprogramme für die Zahleneingabe im Satz ohne Kennzeichen sind über feste Ansprungzellen erreichbar. Die folgenden vier Eingabeprogramme können auf den Zellen 7564 bis 7567 angesprungen werden.

	Ansprungzelle
(a) Eingabe von Zahlen in Gleitkommadarstellung im Satz	7567
(b) Eingabe von Zahlen in Dezimalschreibweise im Satz (vgl. $K_1 = 8$) (Die eingegebenen Zahlen werden intern in Gleitkommazahlen umgewandelt.)	7566
(c) Eingabe von Zahlen in Dezimalschreibweise im Satz (vgl. $K_1 = 9$) (Die eingelesenen Zahlen werden in Festkommazahlen, K. h. konvertiert.)	7565
(d) Eingabe von Zahlen in Dezimalschreibweise mit Kommainformation im Satz (vgl. 2.2.2. (d)). (Die eingegebenen Zahlen werden in Festkommazahlen, K. h. umgewandelt.)	7564

Bei (d) ist die Angabe des Parameters p_1 notwendig. Er bewirkt, daß alle Zahlen des Satzes mit 10^{p_1} multipliziert werden.

Vor dem Ansprung dieser Eingabeprogramme muß der Abspeicherbefehl durch das Hauptprogramm in der Zelle 7745 aufgebaut werden:

$a_1\, a_2\, a_3\, a_4$ 240.

Dabei gibt $a_1\, a_2\, a_3\, a_4$ die Adresse an, ab der die Speicherung beginnen soll.

Die Rückkehr ins Hauptprogramm ist durch das Schlußzeichen möglich. Im Fernschreib-Code wird als Schlußzeichen eines Satzes das Zeichen Doppelpunkt „:" verwendet. Im R 300-Code kann ein Satz sowohl durch das Zeichen „Satzmarke/WRZL" als auch durch das Zeichen „Blockmarke (BM)" abgeschlossen werden. Ein Unterschied ergibt sich aus der Wirkung der beiden Zeichen.

Wenn man annimmt, daß der Befehl „Sprung mit Rückkehrabsicht zur Satzeingabe" in der Zelle n des Hauptprogrammes steht, so wird bei Eingabe von „SM" zur Zelle n + 1 zurückgesprungen, während bei Eingabe von „BM" die Rückkehr zur Zelle n + 2 erfolgt. In allen im Abschnitt 2.2. behandelten Eingabeunterprogrammen sind keine Anwahlbefehle berücksichtigt.

2.2.4. Bemerkungen zur Eingabe

Bei Satzeingabe im R 300-Code hat das Zeichen Doppelpunkt „:" dieselbe Wirkung wie das Zeichen Blockmarke „BM".

Die Satzschlußzeichen „SM/WRZL", „BM" und „:" schließen die Funktion der Wortmarke nicht ein, d. h. vor der Satzmarke ist die Marke des letzten Wortes anzugeben.

Ein Wort und ein Satz können stets gelöscht werden, solange nicht die entsprechenden Schlußzeichen eingegeben sind.

Das Löschen einer Zahl, eines Befehls, eines Externbefehls oder einer Adresse kann durch das Zeichen Zwischenraum „ZR" geschehen, während das Löschen eines Satzes in der 8-Kanal-Version durch die Zeichen „Irrung Satz (IS)", „Irrung Block (IB)" und Apostroph „'" und in der 5-Kanal-Version durch das Zeichen „H" erreicht werden kann. Das Irrungszeichen kann an einer beliebigen Stelle des Satzes vor dem Satzschlußzeichen gegeben werden. Nach dem Löschen eines Wortes oder eines Satzes kann das Wort bzw. der Satz neu eingegeben werden. Die erneute Satzeingabe ist in jedem Fall auch bei der Eingabe mit Kennzeichen $K_1 = 8$ oder $K_1 = 9$ mit der ersten Zahl zu beginnen, während ein Wort bei der Eingabe mit Kennzeichen beginnend mit dem ersten Kennzeichen neu eingegeben werden muß.

Im 5-Kanal-Code hat das Auftreten der Zeichen Wagenrücklauf „WR", Zeilenvorschub „ZV", Ziffern/Zeichen „Zi/Zei" und Buchstaben „Bu" zwischen zwei Zahlen keine Wirkung. Dies gilt im 8-Kanal-Code nur für die Zeichen „WR", Großbuchstaben „GB" und Kleinbuchstaben „KB". Tritt hier „ZV" auf, so ist anschließend mit „ZR" zu löschen.

3. Rechenoperationen

Da beim Cellatron C 8205 an Rechenoperationen nur die Festkommaaddition bzw. -subtraktion verdrahtet sind, müssen die Festkommamultiplikation und -division sowie alle Gleitkommarechenoperationen über Unterprogramme realisiert werden. Diese Standardprogramme für die Festkommamultiplikation und -division, die Gleitkommaaddition und -subtraktion, die Gleitkommamultiplikation und -division sind im BIS als Bausteine enthalten. Die Parameterversorgung für diese Unterprogramme wird besonders geregelt. Da der C 8205 außer dem Akkumulator im Rechenwerk kein weiteres Register besitzt, sind für die Parameterversorgung der offenen Unterprogramme verschiedene „Register" definiert worden. Es handelt sich dabei um drei Speicherzellen auf der letzten Spur, und zwar:

Register R_1: Zelle 7773

Register R_2: Zelle 7776

Register R_3: Zelle 7774

Ihre Anwendung bei den einzelnen Programmen ist an bestimmte Regeln gebunden, die vom Programmierer beachtet werden müssen.

Vor dem Aufruf eines Unterprogrammes sind die Operanden durch das Hauptprogramm in die erwähnten Register abzuspeichern. Der Aufruf des Unterprogrammes braucht nicht direkt nach der Speicherung der Operanden zu erfolgen, wenn man beachtet, daß die Inhalte der entsprechenden Register inzwischen nicht zerstört werden.

Damit der Programmierer sich bei der Abspeicherung der Bausteine auf verschiedene Speicherplätze nicht jeweils neue Einsprungadressen zu merken braucht, wurden auch hier Ansprungzellen festgelegt. Es werden dazu die Zellen 7540 bis 7553 verwendet. Für die Ergebnisübermittlung gibt es grundsätzlich zwei Möglichkeiten:

— das Ergebnis wird dem Hauptprogramm nur im Akkumulator übermittelt

— das Ergebnis wird dem Hauptprogrammm im Akkumulator und im Register R_1 übermittelt

Wird bei den Gleit- oder Festkommarechnungen der interne Zahlenbereich überschritten, so wird im jeweiligen Unterprogramm der Zustand Überlauf erzeugt und der Rechner geht in den Zustand „unbedingter Stopp". Der Adreßteil des unbedingten Stoppbefehls, der mit Hilfe der Befehlsregisteranzeige kontrolliert werden kann, gibt Auskunft, in welchem Rechenprogramm die Überschreitung des Zahlenbereiches eingetreten ist (vgl. Bedienungsanleitung). In der Adressenanzeige wird die Ansprungzelle des Programms angegeben. Wird z. B. bei der Multiplikation zweier Gleitkommazahlen das Produkt größer als $z = (1-2^{-26}) \cdot 2^{31}$, so wird die Ansprungadresse des Rechenprogramms „Gleitkommamultiplikation" – 7546 – angezeigt.

Will man sehen, welcher Unterprogramm-Aufruf des Hauptprogramms abgearbeitet wurde, kann man mit Hilfe eines Programmes (z. B. Bereichsausschreiben/D3) die gespeicherten Rücksprungadressen ausschreiben.

Bei Unterschreitung des Wertebereiches wird das Ergebnis zu Null definiert.

3.1. Gleitkommaoperationen

3.1.1. Addition, Subtraktion

Die Gleitkommaaddition und -subtraktion benutzen als Operandenregister R_1 und R_2.

Die bei der Addition und Subtraktion entsprechend der Ergebnisübermittlung möglichen vier Varianten können über die folgenden Ansprungzellen erreicht werden:

	Ansprungzelle
(a) Gleitkommaaddition mit Speicherung	7553
(b) Gleitkommaaddition ohne Speicherung	7552
(c) Gleitkommasubtraktion mit Speicherung	7551
(d) Gleitkommasubtraktion ohne Speicherung	7550

Bei den vier Varianten wird im einzelnen gebildet:

(a) $\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle + \langle R_2 \rangle$

(b) $\langle Ac \rangle := \langle R_1 \rangle + \langle R_2 \rangle$

(c) $\langle R_1 \rangle := \langle Ac \rangle := \langle R_2 \rangle - \langle R_1 \rangle$

(d) $\langle Ac \rangle := \langle R_2 \rangle - \langle R_1 \rangle$

Bei der Subtraktion wird so verfahren, daß zunächst der Inhalt von R_1 mit (—1) multipliziert wird und danach $\langle R_1 \rangle$ und $\langle R_2 \rangle$ addiert werden.

Nach der Rechnung stehen im Fall:

(a) und (c) in R_2 der Ausgangswert, in R_1 das Ergebnis;

(b) in R_1 und R_2 die Ausgangswerte;

(d) in R_2 der Ausgangswert, in R_1 der mit (—1) multiplizierte Ausgangswert.

Bei Überschreitung des Zahlenbereiches leuchten die Lampen „FÜ" und „US" und in der Befehlsregisteranzeige wird ein unbedingter Stoppbefehl mit der Adresse 7553 angegeben.

Die Rücksprungadresse ist für die

— Addition und Subtraktion mit Speicherung in der Zelle 7710

— Addition und Subtraktion ohne Speicherung in der Zelle 7707

gespeichert.

Bei den Programmen mit Speicherung steht in der Zelle 7707 der Befehl „7773 240" (Transport nach Zelle 7773).

3.1.2. Multiplikation, Division

Die Gleitkommamultiplikation und -division benutzen als Versorgungszellen die Register R_1 und R_3.
Wie bei der Addition und Subtraktion sind auch hier vier Varianten möglich. Sie können wie folgt angesprungen werden:

	Ansprungzelle
(a) Gleitkommamultiplikation mit Speicherung	7547
(b) Gleitkommamultiplikation ohne Speicherung	7546
(c) Gleitkommadivision mit Speicherung	7545
(d) Gleitkommadivision ohne Speicherung	7544

Bei den vier Varianten wird gebildet:

(a) $\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle \times \langle R_3 \rangle$

(b) $\langle Ac \rangle := \langle R_1 \rangle \times \langle R_3 \rangle$

(c) $\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle : \langle R_3 \rangle$

(d) $\langle Ac \rangle := \langle R_1 \rangle : \langle R_3 \rangle$

Die Ausgangswerte der Register R_1 und R_3 bleiben in keinem Fall erhalten.

Eine Wertebereichsüberschreitung wird durch das Leuchten von „FÜ" und „US" angezeigt.

In der Adressenanzeige wird bei der

— Gleitkommamultiplikation die Adresse 7546

— Gleitkommadivision die Adresse 7544

angegeben.

Die Rücksprungadressen sind für alle vier Varianten in der Zelle 7766 gespeichert.

3.2. Festkommamultiplikation, Festkommadivision

Bei den Festkommaoperationen wird berechnet
$c = a \cdot b \cdot 10^{-p_1}$ bzw. $c = a : b \cdot 10^{p_1}$

Die Größen a, b, c sind Festkommazahlen, K. h. und p_1 ist ein Korrekturfaktor. Sowohl bei der Multiplikation als auch bei der Division ist dieser Korrekturfaktor anzugeben. Für ihn gilt:

$$0 \leqq p_1 \leqq 7; \; p_1 - \text{ganzzahlig} .$$

Der Korrekturfaktor steht in den unbesetzten Bits z_0 z_1 z_2 des Befehls 754x 025. (Sprung mit Rückkehrabsicht zum Rechenprogramm.) Mit Hilfe von p_1 kann bei der Multiplikation und Division eine gedachte Kommastellung realisiert werden. Bei der Division hängt die Genauigkeit des Ergebnisses von p_1 ab.

Die vier möglichen Varianten sind wie folgt anzuspringen:

		Ansprungzelle
(a)	Festkommamultiplikation mit Speicherung	7543
(b)	Festkommamultiplikation ohne Speicherung	7542
(c)	Festkommadivision mit Speicherung	7541
(d)	Festkommadivision ohne Speicherung	7540

Bei den Varianten wird gebildet:

(a) $\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle \times \langle R_3 \rangle \times 10^{-p_1}$
(b) $\langle Ac \rangle := \langle R_1 \rangle \times \langle R_3 \rangle \times 10^{-p_1}$
(c) $\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle : \langle R_3 \rangle \times 10^{p_1}$
(d) $\langle Ac \rangle := \langle R_1 \rangle : \langle R_3 \rangle \times 10^{p_1}$

Im Fall (a) und (c) bleibt der Inhalt des Registers R_3 erhalten, während in R_1 das Ergebnis steht.

Im Fall (b) und (d) werden die Inhalte der Register R_1 und R_3 nicht verändert.

Der Zahlenbereich ist bei beiden Operationen durch

$$|z| \leqq 2^{31} - 1$$

beschränkt.

Eine Wertebereichsüberschreitung wird durch das Leuchten von „FÜ" und „US" angezeigt.

In der Befehlsregisteranzeige werden
— bei der Multiplikation die Adresse 7542
— bei der Division die Adresse 7540
angegeben.

Die Rücksprungadressen sind bei den Varianten ohne Speicherung in der Zelle 7741 und bei den Varianten mit Speicherung in der Zelle 7742 gespeichert. Bei der Multiplikation und Division mit Speicherung steht in der Zelle 7741 der Befehl 7773 240. (Transport nach Zelle 7773.)

4. Ausgabe

Da beim Cellatron C 8205 nur die Ausgabe eines einzigen Zeichens verdrahtet ist, wurden für die Ausgabe von ganzen Worten Standardprogramme geschaffen.

Dabei kann die auszugebende Information verschieden interpretiert werden. Alle Ausgabeunterprogramme sind über feste Ansprungzellen erreichbar. Außer für die im Minimalprogramm enthaltenen Ausgabeprogramme befinden sich diese Zellen auf der Bahn 7540. Es werden die Zellen 7554 bis 7563 verwendet.

Als Versorgungszelle wird bei allen Ausgabeunterprogrammen das Register R_1 benutzt. Der Aufruf des Unterprogramms braucht nicht direkt nach der Speicherung des auszugebenden Wortes nach R_1 erfolgen, wenn man beachtet, daß der Inhalt dieses Registers inzwischen nicht zerstört wird.

Der Inhalt des Akkumulators beim Rücksprung ist belanglos. Durch die Abarbeitung eines Ausgabeprogrammes wird der Inhalt von R_1 nicht verändert. In den Ausgabeunterprogrammen sind keine Anwahlbefehle berücksichtigt.

4.1. Ausgabeprogramme des Minimalprogramms

In das Minimalprogramm wurden drei häufig benutzte Programme zum Druck von Adressen, Befehlen und Triaden aufgenommen. Diese Druckprogramme sind als Unterprogramme verwendbar. Sie können auf den folgenden Zellen angesprungen werden:

	Ansprungzelle
UP Adressendruck	100
UP Befehlsdruck	53
UP Triadendruck	276

4.1.1. Befehls- und Adressendruck

Adressendruck

Beim Adressendruck werden nur die Wortstellen z_{12} bis z_{23} des Maschinenwortes gedruckt. Diese 12 Dualstellen werden mit 4 Oktalziffern ausgegeben. Je drei duale Wortstellen werden zu einer Oktalziffer zusammengefaßt.

Beispiel: 0000 kleinste mögliche Adresse
7777 größte mögliche Adresse

Befehlsdruck

Es werden bei diesem Druck die Wortstellen z_0 bis z_{11} nicht berücksichtigt, da diese für die interne Befehlsinterpretation keine Bedeutung haben.

Die Befehlsadresse wird mit 4, der Operationsteil mit 3 Oktalziffern gedruckt. Zwischen dem Druck der Adresse und des Operationsteils wird ein Leerschritt ausgeführt.

Beispiel: 0172 211
7645 442

4.1.2. Triadendruck

(Ausgabe von Oktalzahlen)

Die Ausgabe des internen Wortes erfolgt hierbei in Form von 11 Oktalziffern. Nach dem Druck von jeweils 4 Oktalziffern wird ein Leerschritt ausgeführt. Damit erhält das Druckbild die Gestalt:

Beispiel: $t_{10}\, t_9\, t_8\, t_7 \quad t_6\, t_5\, t_4\, t_3 \quad t_2\, t_1\, t_0$
7 0 2 6 4 1 3 5 0 7 1 .

Ist in dem internen Maschinenwort das erste Bit z_0 besetzt, so ist die erste ausgedruckte Ziffer $t_{10} \geqq 4$.

4.2. Gleitkommaausgabe

Bei den Programmen zur Gleitkommaausgabe wird der Inhalt des Registers R_1 als Zahl in Gleitkommadarstellung interpretiert.

4.2.1. Ausgabe in Gleitkommadarstellung

Das Programm wird auf der Zelle 7562 angesprungen.

Mit Hilfe dieses Druckprogramms wird der Inhalt von R_1 als Zahl in Gleitkommadarstellung ausgegeben.

Das Druckbild umfaßt 11 Zeichen:

— das Mantissenvorzeichen (bei positivem Vorzeichen wird ein Leerschritt ausgeführt)
— 7 dezimale Mantissenziffern
— Exponentenvorzeichen (bei positivem Vorzeichen wird ein Leerschritt ausgeführt)
— 2 Ziffern für den Dezimalexponenten

$$(—)\ m_1\, m_2\, m_3\, m_4\, m_5\, m_6\, m_7\quad (—)\ e_1\, e_2\ .$$

Der Ausgabezahlenbereich stimmt mit dem internen überein. Gedruckt wird der echte, d. h. nicht erhöhte Dezimalexponent. Er kann maximal den Wert $e_1\, e_2 = 10$ annehmen.

Der Gleitkommanull (intern alle Bits gleich Null) ist eindeutig die Ausgabeschreibweise

0000000—09

zugeordnet.

4.2.2. Ausgabe in Dezimalschreibweise

Durch dieses Ausgabeprogramm wird das als Gleitkommazahl interpretierte interne Maschinenwort nach der Rückkonvertierung als Zahl in Dezimalschreibweise gedruckt.

Von dem Programm gibt es zwei Ausführungen. Sie gestatten die Ausgabe mit und ohne Runden. Die beiden Varianten können über die Zellen 7560 und 7561 erreicht werden.

	Ansprungzelle
(a) Ausgabe einer Gleitkommazahl in Dezimalschreibweise ohne Runden	7560
(b) Ausgabe einer Gleitkommazahl in Dezimalschreibweise mit Runden	7561

Die Anzahl der Zeichen, die gedruckt werden, ist abhängig von zwei Parametern. Beim Ansprung der beiden Druckprogramme ist in den unbesetzten Bits des Befehls „Sprung mit Rückkehrabsicht nach 7560 bzw. 7561" anzugeben, wieviel Stellen vor und nach dem Komma ausgegeben werden sollen. Der Parameter p_2, der die Anzahl der Stellen vor dem Komma festlegt, steht in den Bits $z_4\ z_5\ z_6\ z_7$ des Einsprungbefehls. Der Parameter p_1 bestimmt die Anzahl der Stellen nach dem Komma. Er ist in den Bits $z_8\ z_9\ z_{10}\ z_{11}$ anzugeben. Für beide Parameter gilt:

$$0 \leqq p_1\ ,\ p_2 \leqq 15\ .$$

(Führende Nullen werden als „Zwischenraum" ausgegeben.)

Wenn die zu druckende Zahl mehr Stellen vor dem Komma besitzt, als p_2 angibt, wird „zg" („Zahl zu groß") ausgegeben.

Ein positives Vorzeichen wird nicht gedruckt, während ein negatives Vorzeichen hinter der Zahl steht.

Der Ausgabezahlenbereich stimmt wie bei der Ausgabe in Gleitkommadarstellung mit dem internen Zahlenbereich überein.

Beispiel: Es ist die Zahl —547,13 auszugeben.

Für die Parameter gilt: $p_2 = 3$, $p_1 = 2$.

Druckbild: 547,13—

Gilt für die Parameter $p_2 = 5$, $p_1 = 3$, so hat das Druckbild die Gestalt:

xx547,130—

(vor dem Druck der 1. Ziffer (5) werden zwei Leerschritte ausgeführt, da $p_2 = 5$ ist).

Ist $p_2 = 1$, $p_1 = 2$ angegeben, so wird „zg" gedruckt.

4.3. Festkommaausgabe

Bei den Programmen zur Ausgabe von Festkommazahlen wird das interne Wort als Dualzahl interpretiert und in eine Dezimalzahl umgewandelt. Der Druck erfolgt in Dezimalschreibweise.

In Abhängigkeit von der Druckaufbereitung unterscheidet man bei den Festkommaausgabeprogrammen zwei Varianten, die als Variante I und Variante II bezeichnet werden. Bei der Variante I sind zwei Parameter notwendig, um die Anzahl der Stellen vor und nach dem Komma festzulegen. Außerdem setzt das Programm voraus, daß die Zahl vor der Ausgabe kommagerecht aufbereitet wurde.

Diese spezielle Aufbereitung der Zahl wird bei Variante II nicht gefordert. Dafür sind drei Parameter anzugeben. Zwei Parameter legen die Anzahl der Stellen vor und nach dem Komma fest. Der dritte Parameter gibt an, ab welcher Ziffer der zehnstelligen Festkommazahl der Druck erfolgen soll.

Die Parameter bestimmen auch die Anzahl der zu druckenden Zeichen. Das negative Vorzeichen wird am Ende einer Zahl gegeben. Bei einem positiven Vorzeichen wird ein Leerschritt ausgeführt.

Der Ausgabezahlenbereich ist gleich dem internen, d. h. es können Zahlen aus dem Bereich

$$|z| \leqq 2^{31} - 1 = 2\ 147\ 483\ 647$$

gedruckt werden.

4.3.1. Variante I

Dieses Ausgabeprogramm kann auf der Zelle 7557 angesprungen werden.

Im Einsprungbefehl werden in den Bits $z_4\ z_5\ z_6\ z_7$ die Stellen vor dem Komma (p_2) und in den Bits $z_8\ z_9\ z_{10}\ z_{11}$ die Stellen nach dem Komma (p_1) angegeben.

Für die Parameter gilt:

$$0 \leqq p_1,\ p_2 \leqq 15$$

Vor dem Ansprung des Druckprogramms ist die auszugebende Zahl kommagerecht aufzubereiten. Der Begriff der kommagerechten Aufbereitung soll an einem Beispiel erläutert werden. Die maximal zehn ausgebbaren Dezimalziffern seien wie folgt durchnumeriert:

$$d_9\ d_8\ d_7\ d_6\ d_5\ d_4\ d_3\ d_2\ d_1\ d_0$$

Der Parameter p_2 bestimmt, wieviel Ziffern von d_9 ab beim Druck der Stellen vor dem Komma zu berücksichtigen sind. Der Parameter p_1 gibt an, wieviel Ziffern nach dem Druck des Kommas ausgegeben werden. Ist $p_2 = 5$ und $p_1 = 2$, so werden die Ziffern $d_9\ d_8\ d_7\ d_6\ d_5$ für den Druck der Stellen vor dem Komma und die Ziffern $d_4\ d_3$ für den Druck der Stellen nach dem Komma berücksichtigt.

Damit die Zahl kommagerecht ausgegeben wird, ist Voraussetzung, daß die gedachte Kommastellung der zu druckenden Zahl der durch den Parameter p_2 festgelegten Kommastellung entspricht. Wenn z. B. die Zahl —347,15 gedruckt werden soll und für die Parameter $p_2 = 5$ und $p_1 = 2$ gilt, so muß intern die Zahl —34715000 aufbereitet sein.

Das Druckbild hat in diesem Fall das folgende Aussehen:

xx347,15 .

(Vor dem Druck der ersten Ziffer (3) werden zwei Leerschritte ausgeführt.)

Steht im Register R_1 die oben angegebene Festkommazahl, K. h.

—34715000

und ist $p_2 = 3$, $p_1 = 2$, so wird

xx3,47—

gedruckt.

Wäre $p_2 = 2$, $p_1 = 1$, so würde

x0,3—

ausgegeben.

4.3.2. Variante II

Von diesem Ausgabeprogramm existieren zwei Ausführungen. Sie können über die Ansprungzellen 7555 und 7556 erreicht werden.

	Ansprungzelle
(a) Ausgabe von Festkommazahlen, K. h. ohne Runden	7555
(b) Ausgabe von Festkommazahlen, K. h. mit Runden	7556

Im Einsprungbefehl sind drei Parameter anzugeben:

1. p_3 = erste zu druckende Ziffer in den Bits: $z_0\ z_1\ z_2\ z_3$
2. p_2 = Anzahl der Stellen vor dem Komma in den Bits: $z_4\ z_5\ z_6\ z_7$
3. p_1 = Anzahl der Stellen nach dem Komma in den Bits: $z_8\ z_9\ z_{10}\ z_{11}$

Wenn die zehn Dezimalziffern wieder von 0 bis 9 durchnumeriert werden, so gibt der Parameter p_3 ($0 \leqq p_3 \leqq 9$) die höchste Ziffer an, die bei der Ausgabe berücksichtigt wird.

Beispiel: Für die Parameter p_2 und p_1 gilt: $p_2 = 5$, $p_1 = 2$. Wenn die interne Festkommazahl, K. h. 34715 in der Form 347,15 ausgegeben werden soll, muß der Parameter $p_3 = 6$ sein.

Ist die Zahl intern mit 34715000 angegeben, muß für den Parameter $p_3 = 9$ gelten ($p_2 = 5$, $p_1 = 2$). In beiden Fällen hat das Druckbild die Gestalt

xx347,15

(Vor dem Druck der ersten Ziffer (3) wird zweimal Zwischenraum ausgegeben.)

Wenn die auszugebende Zahl größer ist, als der Parameter p_3 angibt, d. h. wenn Dezimalziffern vor der durch p_3 bestimmten Ziffer ungleich Null sind, so wird

„zg" (Zahl zu groß)

gedruckt.

Ist z. B. in R_1 die Zahl

—34715000

angegeben und ist $p_3 \leqq 6$, so erfolgt für beliebiges p_2 und p_1 der Druck von „zg".

4.4. Tetradenausgabe

Dieses Ausgabeprogramm interpretiert das interne Maschinenwort als Zahl in Tetradendarstellung. Der Druck erfolgt wie bei Variante II.

Das Programm kann auf der Zelle 7554 angesprungen werden. Im Befehl „Sprung mit Rückkehrabsicht" sind wie bei Variante II die drei Parameter

1. p_3 = erste zu druckende Ziffer in den Bits: $z_0\ z_1\ z_2\ z_3$
2. p_2 = Anzahl der Stellen vor dem Komma in den Bits: $z_4\ z_5\ z_6\ z_7$
3. p_1 = Anzahl der Stellen nach dem Komma in den Bits: $z_8\ z_9\ z_{10}\ z_{11}$

anzugeben. Für den Parameter p_3 gilt $0 \leqq p_3 \leqq 7$.

Der Ausgabezahlenbereich ist durch

$$z \leqq 10^8 - 1$$

beschränkt, d. h. er stimmt mit dem internen Zahlenbereich überein.

Auch bei diesem Programm wird „zg" gedruckt, wenn die auszugebende Zahl größer ist, als der Parameter p_3 festlegt.

Tritt eine Pseudotetrade auf, d. h. einer Tetrade entspricht eine Zahl z mit $10 \leqq z \leqq 15$, so wird p (Pseudotetrade) ausgeschrieben.

4.5. Triadenausgabe auf Lochband

Das Programm ist auf der Zelle 7563 anzuspringen.

Mit Hilfe dieses Programms wird der Inhalt des Registers R_1 in einer Form gestanzt, in der er wieder vom Rechner eingelesen werden kann. Vor dem Stanzen der 11 Triaden werden einmal das Zeichen Zwischenraum „ZR" und die Eingabekennzeichen $K_1 = 0$ und $K_2 = 0$ ausgegeben.

Es gibt von diesem Programm zwei Ausführungen, die die Lochbandausgabe im R 300-Lochband-Code und im FS-Code gestatten.

Bei beiden Ausführungen wird ein Wort durch die Wortmarke Punkt „." abgeschlossen.

5. Externbefehle

Die im BIS enthaltenen Standardprogramme haben in ihrer Anwendung den Nachteil, daß dem Programmierer sowohl die Adressen der benutzten Register als auch die Ansprungadressen der Standardprogramme bekannt sein müssen. Um diese Kenntnis zu umgehen, wurden Pseudobefehle geschaffen, die im weiteren auch als Externbefehle bezeichnet werden.

Die Umwandlung der Externbefehle in echte interne Maschinenbefehle erfolgt bei der Eingabe (vgl. 2.1.2., $K_1 = +$). Dabei wird aus einer auf dem Speicher stehenden Tabelle der einem Externbefehl zugeordnete Internbefehl gesucht und nach eventueller Addition eines Parameterteils abgespeichert. Im gespeicherten Programm treten also keine Pseudobefehle mehr auf.

Mit Hilfe der Oktalziffern O_1O_2 wird der entsprechende Internbefehl bestimmt, d. h. O_1 und O_2 legen die auszuführende Operation fest.

Bei den Externbefehlen mit $O_1O_2 \neq 15, 16, 25, 26$ können nach der Angabe der beiden Operationsziffern bis zu drei Parameter eingegeben werden. Diese Parameter p_i mit

$$0 \leqq p_i \leqq 9 \qquad (i = 1, 2, 3)$$

werden durch Tetraden dargestellt, die die folgenden Dualstellen des durch O_1O_2 festgelegten Maschinenbefehls belegen:

$z_0\ z_1\ z_2\ z_3$	$z_4\ z_5\ z_6\ z_7$	$z_8\ z_9\ z_{10}\ z_{11}$	$z_{12} \ldots\ldots\ldots z_{32}$
p_3	p_2	p_1	Maschinenbefehl

Abb. 7

Bei den Externbefehlen mit $O_1O_2 = 15, 16, 25, 26$ kann nach den beiden Ziffern O_1 und O_2 ein Parameter p_1 mit $0 \leqq p_1 \leqq 7$ angegeben werden. Dieser Parameter wird durch eine Triade dargestellt, die die Bits $z_0\ z_1\ z_2$ des zugehörigen Maschinenbefehls belegt.

Während bei der Angabe der Operationsziffern beide Ziffern gegeben werden müssen, können bei der Eingabe der Parameter führende Nullen weggelassen werden (vgl. 2.1.2., $K_1 = +$).

Werden im ersten Fall ($O_1O_2 \neq 15, 16, 25, 26$) mehr als drei Parameter oder im zweiten Fall ($O_1O_2 = 15, 16, 25, 26$) mehr als ein Parameter bzw. ein Parameter $p_1 \geqq 8$ eingegeben, so gehen die zuerst eingegebenen Parameter verloren bzw. der Parameter wird verfälscht.

Bei der Verwendung von Externbefehlen ist zu beachten, daß nur Parameter p_i mit $0 \leqq p_i \leqq 9$ $(i = 1, 2, 3)$ eingegeben werden können. Tritt ein Parameter auf, der größer als neun ist, so muß bei der Programmierung die interne Darstellung benutzt werden.

Ein Beispiel für die Anwendung von Externbefehlen ist im Abschnitt 9.3. angeführt.

Entsprechend der Operationsziffer O_1 unterscheidet man verschiedene Befehlsgruppen. Die Ziffer O_2 gibt den Befehl innerhalb einer Gruppe an.

5.1. Ubertragungsbefehle ($O_1 = 0$)

Externbefehl	Internbefehl mit Kennzeichen	Funktion
01.	00 7773 240.	$\langle R_1 \rangle := \langle Ac \rangle$
02.	00 7776 240.	$\langle R_2 \rangle := \langle Ac \rangle$
03.	00 7774 240.	$\langle R_3 \rangle := \langle Ac \rangle$
04.	00 7745 240.	$\langle 7745 \rangle := \langle Ac \rangle$

5.2. Rechenoperationen ohne Speicherung ($0_1 = 1$)

Externbefehl	Internbefehl mit Kennzeichen	Funktion
11.	00 7552 025.	$\langle Ac \rangle := \langle R_1 \rangle + \langle R_2 \rangle$ (Gleitkommaaddition)
12.	00 7550 250.	$\langle Ac \rangle := \langle R_2 \rangle - \langle R_1 \rangle$ (Gleitkommasubtraktion)
13.	00 7546 025.	$\langle Ac \rangle := \langle R_1 \rangle \times \langle R_3 \rangle$ (Gleitkommamultiplikation)
14.	00 7544 025.	$\langle Ac \rangle := \langle R_1 \rangle : \langle R_3 \rangle$ Gleitkommadivision)
15 p_1.	00 $p_1$000 7542 025.	$\langle Ac \rangle := \langle R_1 \rangle \times \langle R_3 \rangle \times 10^{-p_1}$ (Festkommamultiplikation)
16 p_1.	00 $p_1$000 7540 025.	$\langle Ac \rangle := \langle R_1 \rangle : \langle R_3 \rangle \times 10^{p_1}$ (Festkommadivision)

5.3. Rechenoperationen mit Speicherung ($0_1 = 2$)

Externbefehl	Internbefehl mit Kennzeichen	Funktion
21.	00 7553 025.	$\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle + \langle R_2 \rangle$ (Gleitkommaaddition)
22.	00 7551 025.	$\langle R_1 \rangle := \langle Ac \rangle := \langle R_2 \rangle - \langle R_1 \rangle$ (Gleitkommasubtraktion)
23.	00 7547 025.	$\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle \times \langle R_3 \rangle$ (Gleitkommamultiplikation)
24.	00 7545 025.	$\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle : \langle R_3 \rangle$ (Gleitkommadivision)
25 p_1.	00 $p_1$000 7543 025.	$\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle \times \langle R_3 \rangle \times 10^{-p_1}$ (Festkommamultiplikation)
26 p_1.	00 $p_1$000 7541 025.	$\langle R_1 \rangle := \langle Ac \rangle := \langle R_1 \rangle : \langle R_3 \rangle \times 10^{p_1}$ (Festkommadivision)

5.4. Ausgabebefehle ($0_1 = 3$ und $0_1 = 4$)

Externbefehl	Internbefehl mit Kennzeichen	Funktion
31.	00 276 025.	Druck von $\langle R_1 \rangle$ als 11stellige Oktalzahl
32.	00 100 025.	Druck von $\langle R_1 \rangle$ als 4stellige Adresse
33.	00 53 025.	Druck von $\langle R_1 \rangle$ als Befehl
34.	00 7562 025.	Druck von $\langle R_1 \rangle$ in Gleitkommadarstellung
35 p_2p_1.	00 $t_9t_8t_7$ 7560 025.	Druck von $\langle R_1 \rangle$ als Gleitkommazahl in Dezimalschreibweise ohne Runden
36 p_2p_1.	00 $t_9t_8t_7$ 7561 025.	Druck von $\langle R_1 \rangle$ als Gleitkommazahl in Dezimalschreibweise mit Runden
41.	00 7563 025.	Stanzen von $\langle R_1 \rangle$ als 11stellige Oktalzahl mit $K_1 = 0$ und $K_2 = 0$
44 p_2p_1.	00 $t_9t_8t_7$ 7557 025.	Druck von $\langle R_1 \rangle$ als Festkommazahl in Dezimalschreibweise ohne Runden (Variante I)
45 $p_3p_2p_1$.	00 $t_{10}t_9t_8t_7$ 7555 025.	Druck von $\langle R_1 \rangle$ als Festkommazahl in Dezimalschreibweise ohne Runden (Variante II)

Externbefehl	Internbefehl mit Kennzeichen	Funktion
46 $p_3p_2p_1$.	00 $t_{10}t_9t_8t_7$ 7556 025.	Druck von $\langle R_1 \rangle$ als Festkommazahl in Dezimalschreibweise mit Runden (Variante II)
47 $p_3p_2p_1$.	00 $t_{10}t_9t_8t_7$ 7554 025.	Druck von $\langle R_1 \rangle$ als Zahl in Tetradendarstellung in Dezimalschreibweise ohne Runden (Variante II)

5.5. Eingabebefehle ($0_1 = 5$ und $0_1 = 6$)

Externbefehl	Internbefehl mit Kennzeichen	Funktion
51.	00 262 025.	Eingabe eines Wortes mit Kennzeichen
52.	00 316 025.	Eingabe einer Adresse
53.	00 263 025.	Eingabe eines Programmstückes
54.	00 7575 025.	Eingabe einer Gleitkommazahl ohne Kennzeichen
55.	00 7574 025.	Eingabe einer Dezimalzahl ohne Kennzeichen (intern Konvertierung in eine Gleitkommazahl)
56.	00 7567 025.	Eingabe von Gleitkommazahlen im Satz ohne Kennzeichen
57.	00 7566 025.	Eingabe von Dezimalzahlen im Satz ohne Kennzeichen (intern Konvertierung in Gleitkommazahlen)
61.	00 7573 025.	Eingabe einer Dezimalzahl ohne Kennzeichen (intern Konvertierung in eine Festkommazahl, Komma hinten)
62.	00 7570 025.	Eingabe einer positiven Dezimalzahl ohne Kennzeichen (intern Konvertierung in Tetraden)
63.	00 7571 025.	Eingabe einer maximal 8-stelligen Dezimalzahl ohne Kennzeichen (intern Konvertierung in eine Festkommazahl, Komma hinten)
64 p_1.	00 t_8t_7 7572 025.	Eingabe einer Dezimalzahl mit Kommainformation ohne Kennzeichen (intern Konvertierung in eine Festkommazahl, Komma hinten)
65.	00 7565 025.	Eingabe von Dezimalzahlen im Satz ohne Kennzeichen (intern Konvertierung in Festkommazahlen, Komma hinten)
66 p_1.	00 t_8t_7 7564 025.	Eingabe von Dezimalzahlen mit Kommainformation im Satz ohne Kennzeichen (intern Konvertierung in Festkommazahlen, Komma hinten)

5.6. Anwahlbefehle und Schreibmaschinenfunktionen ($0_1 = 7$)

Externbefehl	Internbefehl mit Kennzeichen	Funktion
70.	00 4700 030.	Anwahl Kanal 0
71.	00 4600 030.	Anwahl Kanal 1
72.	00 4500 030.	Anwahl Kanal 2
73.	00 2720 030.	Wagenrücklauf mit Zeilenvorschub
74.	00 1720 030.	Zeilenvorschub
75.	00 1550 030.	Tabulator
76.	00 320 030.	Zwischenraum

Eine ausführliche Beschreibung der aufgeführten Internbefehle sowie der Wirkungsweise der Parameter ist in den Abschnitten 2.2., 3. und 4. dieses Heftes bzw. in dem Heft „Anleitung zur Programmierung in der Maschinensprache" angegeben.

6. Serviceprogramme

Serviceprogramme sind für die Testung von Programmen und für die Bedienung des Automaten gedacht.

Bei der Abarbeitung eines getesteten Programms werden diese Programme nicht benötigt, so daß sie dann nicht auf dem Speicher zu stehen brauchen. Alle Serviceprogramme werden mittels der ersten Kennzeichen „B" und „D" angesprungen. Die Unterscheidung erfolgt durch ein zweites Kennzeichen.

6.1. Druck von ⟨7777⟩

Mit Hilfe der Eingriffstasten (HGÜE) kann der Inhalt des Akkumulators nach der Speicherzelle 7777 transportiert werden. Durch anschließendes Drücken von D0, D1, D2 wird der Inhalt von 7777 entweder in Triadendarstellung, als Gleitkommazahl oder ganze Dezimalzahl ausgeschrieben. Dabei können D0, D1, D2 in beliebiger Reihenfolge hintereinander ausgeführt werden, da durch den Druck der Inhalt von 7777 nicht verändert wird. Bei Verwendung von D1 muß das UP „Gleitkommaausgabe", bei Benutzung von D2 das UP „Festkommaausgabe, Variante I" abgespeichert sein.

6.2. Bereichsausschreiben

Es können Teile von Programmen oder auch ganze Pogramme, die sich auf dem Speicher befinden, ausgeschrieben werden (mit Ausnahme der durch die Druckprogramme benutzten Zellen).

Das Programm druckt den Inhalt aller Zellen ab einer Anfangsadresse $a_1 a_2 a_3 a_4$ bis zu einer Endadresse $b_1 b_2 b_3 b_4$ aus. Bei der spaltenweisen Ausgabe wird die Adresse des jeweiligen Speicherplatzes mit ausgeschrieben.

Bereichsausschreiben D3

Nach dem Ansprung des Programms über D3 werden zwei Adressen eingegeben

D3 $a_1 a_2 a_3 a_4$. $b_1 b_2 b_3 b_4$.

Die Ausgabe des Wortes erfolgt als Befehl, wenn die Wortstellen z_0 bis z_{11} nicht besetzt sind, anderenfalls erfolgt sie in Triadendarstellung.

Bereichsausschreiben D4

Hier wird eine weitere Unterscheidung durchgeführt. Wenn das Bit z_7 – das höchste Mantissenbit einer Gleitkommazahl – besetzt ist, so wird der Inhalt des Speicherplatzes als Gleitkommazahl ausgegeben. Ist $z_7 = 0$, so erfolgt der Druck als Befehl oder als 11stellige Oktalzahl.

Das Eingabebild hat hier die Gestalt

D4 $a_1 a_2 a_3 a_4$. $b_1 b_2 b_3 b_4$.

Bei Verwendung von D4 muß das UP „Gleitkommaausgabe" abgespeichert sein.

Bereichsausschreiben D5

Hier werden bei der Ausgabe die Unterscheidungen:

— Festkommazahl
— Befehl
— Oktalzahl

gemacht. Ist das letzte Bit z_{32} nicht besetzt, so wird der Inhalt der Speicherzelle als ganze Dezimalzahl ausgegeben.

Eingabebild: D5 $a_1 a_2 a_3 a_4$. $b_1 b_2 b_3 b_4$.

Bei Benutzung von D5 muß das UP „Festkommaausgabe, Variante I" abgespeichert sein.

Hat man eine Adresse falsch eingegeben, so kann man mit Zwischenraum löschen. Anschließend ist das Hilfsprogramm erneut durch D3, D4 oder D5 anzuspringen.

6.3. Protokollprogramm

Das Protokollprogramm wird zum Testen eines Programms benötigt. Es arbeitet das vorgegebene Hauptprogramm Befehl für Befehl ab.

Es wird durch D6 oder D7 angesprungen. Bei D6 wird der Inhalt des Akkumulators entweder als Oktalzahl oder als Gleitkommazahl gedruckt. Ist das höchste Mantissenbit z_7 besetzt, so erfolgt die Ausgabe als Gleitkommazahl, ansonsten in Triadendarstellung. Bei D7 wird bei der Ausgabe des Akkumulatorinhaltes zwischen Oktalzahl und Festkommazahl unterschieden. Die Unterscheidung erfolgt mit Hilfe des Bits z_{32}. Ist dieses Bit nicht besetzt, so wird der Akkumulatorinhalt als ganze Dezimalzahl gedruckt. Bei Ansprung des Protokollprogramms durch D6 muß das UP „Gleitkommaausgabe", bei Ansprung durch D7 das UP „Festkommaausgabe, Variante I" abgespeichert sein. Nach dem Ansprung des Protokollprogramms durch D6 oder D7 wird durch Eingabe eines weiteren Zeichens entschieden, ob und welche Unterprogramme mit zu protokollieren sind.

Es bewirkt:

0 — alle UP werden protokolliert

1 — UP werden nicht protokolliert

2 — UP des BIS werden nicht protokolliert

3 — alle UP, die durch den Operationsteil 025 angesprungen werden, werden nicht protokolliert

Danach wird die Adresse des ersten zu protokollierenden Befehls angegeben. Die folgende Eingabe von zwei weiteren Adressen legt fest, von welcher Anfangsadresse bis zu welcher Endadresse der Akkumulatorinhalt auszudrucken ist. Nachdem bis zu dieser Endadresse protokolliert worden ist, geht der Automat in den Zustand „unbedingter Stopp". Wird der unbedingte Stopp übergangen – z. B. mit Hilfe der Starttaste (vgl. Bedienungsanleitung) –, kann durch Eingabe der Kennzeichen „D" oder „B" entschieden werden, ob der Inhalt von Speicherzellen ausgedruckt oder ob das Protokollieren fortgesetzt werden soll. Falls das Kennzeichen „D" eingegeben wird, erfolgt die weitere Eingabe analog zum Bereichsausschreiben.

Das heißt, beim Ausschreiben bewirkt die Angabe von:

D3 — die Unterscheidung: Triadendarstellung — Befehl

D4 — die Unterscheidung: Triadendarstellung — Befehl — Gleitkommazahl

D5 — die Unterscheidung: Triadendarstellung — Befehl — Festkommazahl

Danach kann erneut eines der Kennzeichen „B" oder „D" eingegeben werden.

Wird das Kennzeichen „B" gegeben, so sind anschließend eine neue Anfangs- und eine neue Endadresse anzugeben. In diesem Fall wird das Protokollieren mit Ausgabe des Akkumulatorinhaltes zwischen den beiden neuen Adressen fortgesetzt.

Das Eingabebild erhält damit die Gestalt:

$$D \begin{Bmatrix}6\\7\end{Bmatrix} \begin{Bmatrix}0\\1\\2\\3\end{Bmatrix} a.b_1.c_1. \left\{\begin{matrix} D \text{ Eingabebild Bereichsausschreiben} \\ B \; b_2.c_2. \end{matrix}\right\} \begin{Bmatrix}D\\B\end{Bmatrix} \text{usw.}$$

Dabei sind:

a — Adresse des ersten zu protokollierenden Befehls
b — Adresse ab der die Ausgabe des Akkumulatorinhaltes erfolgt
c — Adresse bis zu der die Ausgabe des Akkumulatorinhaltes erfolgt

Das Protokollprogramm druckt aus:

— bei Sprungbefehlen:	Zieladresse des Sprunges und Akkumulatorinhalt nach Ausführung des Sprunges
— bei Anwahlbefehlen:	AW K 0 = Anwahl von K0 K = 1 = Anwahl von K1 2 = Anwahl von K2
— bei Eingabebefehlen:	EG B (B = Adresse der Eingabebahn)
— bei X-Befehlen:	X
— bei Sprungbefehlen mit Rückkehrabsicht:	UP a (a = Zieladresse des Sprungbefehls)
— bei Überlauf:	a (a = Adresse des Befehls, bei dem der Überlauf eingetreten ist)

Bei Druckbefehlen werden nur Kleinbuchstaben berücksichtigt und nach jedem Druckbefehl wird der Akkumulatorinhalt ausgegeben.

Bei bedingten Sprungbefehlen mit Rückkehrabsicht erfolgt keine Rückkehr zum Hauptprogramm, wenn der Sprung ausgeführt wird. (Bedingung: Schalter „SP" gesetzt)

Nach dem Druck von EG B und X bei Eingabe- bzw. X-Befehlen geht der Rechner in unbedingten Stopp. Das Protokollieren kann mit Hilfe der folgenden Tastenkombination fortgesetzt werden:

„H" „Starttaste" E E E E G Ü E G M ↓ E M ↑ „SP" setzen R

Es wird die Adresse des nächsten abzuarbeitenden Befehls und der durch den Eingabe- bzw. X-Befehl entstandene Akkumulatorinhalt in Triadendarstellung ausgegeben.

6.4. Bereichsausstanzen

Nach dem Ansprung des Programms durch D8 sind zwei Adressen $a_1\, a_2\, a_3\, a_4$ und $b_1\, b_2\, b_3\, b_4$ einzugeben. Es wird der Inhalt aller Zellen ab $a_1\, a_2\, a_3\, a_4$ bis $b_1\, b_2\, b_3\, b_4$ ausgestanzt. Die Ausgabe erfolgt in Triadendarstellung mit $K_1 = 0$ und $K_2 = 0$. Nach dem Stanzen von 11 Triaden wird das Zeichen „Punkt" als Wortmarke ausgegeben. Zwischen zwei Worten wird zweimal „ZR" gestanzt. Die Hauptleitadresse wird als Anfangsadresse mit 21 $a_1\, a_2\, a_3\, a_4$. gelocht.

Nach Ausgabe des Inhaltes der durch die Endadresse angegebenen Speicherzelle wird das Lochbandschlußzeichen „/" (Schrägstrich) gestanzt.

Eingabebild: D8 $a_1\, a_2\, a_3\, a_4$. $b_1\, b_2\, b_3\, b_4$.

Bei Eingabe einer falschen Adresse kann mit Zwischenraum gelöscht und das Stanzprogramm mit D8 neu angesprungen werden.

Die Ausgabe kann sowohl im 5-Kanal-Fernschreib-Code als auch im 8-Kanal-R-300-Code erfolgen.

6.5. Vergleichsprogramm

Das Programm kann mit B0 oder einem Sofort-Sprungbefehl zur Anfangsadresse erreicht werden.

Nach dem Ansprung des Programms ist das Eingabegerät manuell anzuwählen.

Mit Hilfe des Vergleichsprogramms wird eine über die Tastatur, das Schreibwerk oder die Lochbandleser eingegebene Information mit der zugehörigen im Speicher stehenden Information verglichen. Stimmen

die Werte nicht überein, werden die Adresse des Speicherplatzes, bei dem der Unterschied auftrat, die zugehörige Information auf dem Lochband und der Inhalt der Zelle in Triadendarstellung gedruckt:

Adresse	Information auf dem Lochband	Information auf dem Speicher

Das Vergleichsprogramm speichert die eingelesenen Informationen nicht ab. Es verarbeitet nur Worte mit Eingabekennzeichen. Wenn die Hilfsadressen (z. B. HLA) nicht auf dem Lochband stehen, so müssen sie vor der Anwahl des Eingabegerätes eingegeben werden.

Wird der zu vergleichende Lochstreifen über Leser 1 eingelesen, so hält der Rechner nach jeder Ausgabe an. In diesem Fall ist durch die Tastenkombination

H K1 R

wieder Kanal 1 anzuwählen. Durch die Verwendung von Leser 2 kann dieser Nachteil umgangen werden.

7. Bedienungsanleitung

7.1. Aufbau der Bausteine

Das Bausteininterpretiersystem umfaßt 19 Bausteine:

— das Minimalprogramm
— 5 Eingabebausteine
— 5 Ausgabebausteine
— 5 Rechenbausteine
— 3 Servicebausteine

Vom Minimalprogramm (B 1.1.) und den fünf Eingabebausteinen sowie dem Baustein zur Ausgabe auf Lochband und dem Protokollprogramm gibt es jeweils zwei Ausführungen. Sie ermöglichen die wahlweise Verwendung des 5-Kanal-Fernschreib-Codes und des 8-Kanal-R 300-Codes.

Es ist vorgesehen, das System durch weitere Bausteine zu ergänzen.

7.1.1. Minimalprogramm – B 1.1./5 und B 1.1./8

Das Minimalprogramm bildet den Grundbaustein des Systems. Es belegt die Speicherzellen 0–337. Auf der Bahn Null befindet sich das „Kleine Eingabeprogramm (KEP)".

Außerdem ist im Minimalprogramm das Programm für die Eingabe mit Kennzeichen enthalten. Das Minimalprogramm verarbeitet von den als K_1 zugelassenen Zeichen nur die folgenden:

$K_1 = 0$ Eingabe von Triaden und Befehlen
$K_1 = 1$ Eingabe von Sofortbefehlen
$K_1 = 2$ Eingabe von Hilfsadressen
$K_1 =$ „ZR" (Zwischenraum) Löschen
$K_1 =$ „/" (Schrägstrich) Lochbandschlußzeichen

Für alle übrigen definierten ersten Kennzeichen sind im Minimalprogramm Anschlußbedingungen für die entsprechenden Leseprogramme enthalten.

Die Eingabeunterprogramme:

— Eingabe eines Wortes mit Kennzeichen, Ansprungzelle 262
— Eingabe eines Programmstückes, Ansprungzelle 263
— Eingabe einer Adresse, Ansprungzelle 316

gehören ebenfalls zum Minimalprogramm.

Von den Ausgabeprogrammen wurden als Unterprogramme die Programme:

— Triadendruck	Ansprungzelle 276
— Befehlsdruck	Ansprungzelle 53
— Adressendruck	Ansprungzelle 100

in das Minimalprogramm aufgenommen.

Weiterhin ist im Minimalprogramm das Hilfsprogramm „Druck des Befehlszählers und des Akkumulators" enthalten.

7.1.2. Eingabebausteine

Für die Eingabe von Zahlen in den verschiedenen Darstellungen mit und ohne Kennzeichen sowie die Eingabe von Externbefehlen mit $K_1 = +$ wurden fünf Eingabebausteine geschaffen.

B 2.1./5 und B 2.1./8

Mit diesem Baustein können Gleitkommazahlen und Dezimalzahlen eingegeben werden. Die eingelesenen Dezimalzahlen werden intern in Gleitkommazahlen konvertiert.

Der Baustein besteht aus folgenden Eingabeprogrammen:

(a) Eingabe einer Zahl in Dezimalschreibweise mit $K_1 = 3$

(b) UP Eingabe einer Zahl in Dezimalschreibweise, Ansprungzelle 7574

(c) Eingabe einer Zahl in Gleitkommadarstellung mit $K_1 = 4$

(d) UP Eingabe einer Zahl in Gleitkommadarstellung, Ansprungzelle 7575

(e) Eingabe von Zahlen in Dezimalschreibweise ohne Kennzeichen im Satz mit $K_1 = 8$

(f) UP Eingabe von Zahlen in Dezimalschreibweise im Satz ohne Kennzeichen, Ansprungzelle 7566

(g) UP Eingabe von Zahlen in Gleitkommadarstellung im Satz ohne Kennzeichen, Ansprungzelle 7567

Bei der Eingabe mittels dieses Bausteins werden Konstanten verwendet, die die Spur 340–377 belegen. Die Spur 340–377 wird im folgenden als Konstanteneingabespur bezeichnet. Werden diese Konstanten nicht benötigt, so kann die Spur 340–377 anderweitig verwendet werden.

B 2.2./5 und B 2.2./8

Dieser Baustein dient zur Eingabe von Zahlen in der Darstellung Festkomma hinten. Zu B 2.2. gehören vier Eingabeprogramme:

(a) Eingabe einer Zahl in Dezimalschreibweise (Festkomma, K. h.) mit $K_1 = 5$

(b) Eingabe von Zahlen in Dezimalschreibweise (Festkomma, K. h.) ohne Kennzeichen im Satz mit $K_1 = 9$

(c) UP Eingabe einer Zahl in Dezimalschreibweise (Festkomma, K. h.) ohne Kennzeichen, Ansprungzelle 7573

(d) UP Eingabe von Zahlen in Dezimalschreibweise (Festkomma, K. h.) im Satz ohne Kennzeichen, Ansprungzelle 7565

B 2.3./5 und B 2.3./8

Mit dem Baustein können Festkommazahlen, Komma hinten, und Tetraden eingegeben werden. Er umfaßt die Eingabeprogramme:

(a) Eingabe einer Zahl in Dezimalschreibweise mit maximal 8 Ziffern ohne Vorzeichen mit $K_1 = 6$

(b) UP Eingabe einer Zahl in Dezimalschreibweise mit maximal 8 Ziffern ohne Vorzeichen, Ansprungzelle 7570

(c) Eingabe einer Zahl in Dezimalschreibweise mit maximal 8 Ziffern mit $K_1 = 7$

(d) UP Eingabe einer Zahl in Dezimalschreibweise mit maximal 8 Ziffern, Ansprungzelle 7571

Bei (a) und (b) wird die Zahl intern durch Tetraden dargestellt, während sie bei (c) und (d) in eine Festkommazahl, K. h., konvertiert wird.

Bei Benutzung der Programme (c) und (d) wird die Konstanteneingabespur benötigt.

B 2.4./5 und B 2.4./8

In diesem Baustein sind zwei Eingabeunterprogramme für die Eingabe von Festkommazahlen, K. h., mit Kommainformation zusammengefaßt:

(a) UP Eingabe einer Zahl in Dezimalschreibweise mit Kommainformation (Festkomma, K. h.) ohne Kennzeichen, Ansprungzelle 7572

(b) UP Eingabe von Zahlen in Dezimalschreibweise mit Kommainformation (Festkomma, K. h.) im Satz ohne Kennzeichen, Ansprungzelle 7564

B 2.5./5 und B 2.5./8

Mit Hilfe dieses Bausteins können Externbefehle mit dem ersten Kennzeichen $K_1 = +$ (Plus) eingegeben werden.

Für den Speicherbedarf der Bausteine gilt (oktal):

B 2.1./5 und B 2.1./8	340 Zellen (7 Spuren) + Spur 340–377
B 2.2./5 und B 2.2./8	140 Zellen (3 Spuren)
B 2.3./5 und B 2.3./8	100 Zellen (2 Spuren) + Spur 340–377
B 2.4./5 und B 2.4./8	140 Zellen (3 Spuren)
B 2.5./5 und B 2.5./8	140 Zellen (3 Spuren)

7.1.3. Rechenbausteine

Für die vier Gleitkomma- und zwei Festkommaoperationen gibt es fünf Bausteine.

B 5.1.

Dieser Baustein setzt sich aus den Programmen für die Gleitkommaaddition und -subtraktion zusammen.

(a) UP Gleitkommaaddition mit Speicherung, Ansprungzelle 7553
(b) UP Gleitkommaaddition ohne Speicherung, Ansprungzelle 7552
(c) UP Gleitkommasubtraktion mit Speicherung, Ansprungzelle 7551
(d) UP Gleitkommasubtraktion ohne Speicherung, Ansprungzelle 7550

B 5.2. – Gleitkommamultiplikation

(a) UP Gleitkommamultiplikation mit Speicherung, Ansprungzelle 7547
(b) UP Gleitkommamultiplikation ohne Speicherung, Ansprungzelle 7546

B 5.3. – Gleitkommadivision

(a) UP Gleitkommadivision mit Speicherung, Ansprungzelle 7545
(b) UP Gleitkommadivision ohne Speicherung, Ansprungzelle 7544

B 5.4. – Festkommamultiplikation

(a) UP Festkommamultiplikation mit Speicherung, Ansprungzelle 7543
(b) UP Festkommamultiplikation ohne Speicherung, Ansprungzelle 7542

B 5.5. – Festkommadivision

(a) UP Festkommadivision mit Speicherung, Ansprungzelle 7541
(b) UP Festkommadivision ohne Speicherung, Ansprungzelle 7540

Für den Speicherplatzbedarf gilt im einzelnen (oktal):

B 5.1.	674 Speicherzellen (14 Spuren)
B 5.2.	116 Speicherzellen (3 Spuren)
B 5.3.	176 Speicherzellen (4 Spuren)
B 5.4.	200 Speicherzellen (4 Spuren)
B 5.5.	340 Speicherzellen (7 Spuren)

7.1.4. Ausgabebausteine

Mit Hilfe der fünf Ausgabebausteine können Zahlen in der Gleitkomma-, Festkomma- und Tetradendarstellung ausgegeben werden. Ein Baustein wurde für die Triadenausgabe auf Lochband geschaffen.

B 4.1./5 und B 4.1./8

Es gibt von diesem Baustein zwei Ausführungen. Sie gestatten die Ausgabe auf Lochband im Fernschreib-Code Nr. 2 und im R 300-Lochband-Code.

In den Bausteinen B 4.1./5 und B 4.1./8 sind jeweils zwei Programme für die Lochbandausgabe zusammengefaßt.

(a) UP Stanzen eines Wortes in Triaden, Ansprungzelle 7563

(b) Bereichsausstanzen, Ansprung mit D8

B 4.2.

UP Ausgabe einer Gleitkommazahl in Gleitkommaschreibweise, Ansprungzelle 7562

B 4.3.

(a) UP Ausgabe einer Gleitkommazahl in Dezimalschreibweise ohne Runden, Ansprungzelle 7560

(b) UP Ausgabe einer Gleitkommazahl in Dezimalschreibweise mit Runden, Ansprungzelle 7561

B 4.4.

UP Ausgabe einer Festkommazahl, K. h. in Dezimalschreibweise (Variante I), Ansprungzelle 7557

B 4.5.

(a) UP Ausgabe einer Festkommazahl, K. h. in Dezimalschreibweise ohne Runden (Variante II), Ansprungzelle 7555

(b) UP Ausgabe einer Festkommazahl, K. h. in Dezimalschreibweise mit Runden (Variante II), Ansprungzelle 7556

(c) UP Ausgabe einer Zahl in Tetradendarstellung in Dezimalschreibweise ohne Runden, Ansprungzelle 7554

Die Ausgabebausteine haben den folgenden Speicherplatzbedarf (oktal):

B 4.1./5	100 Zellen	(2 Spuren)
B 4.1./8	40 Zellen	(1 Spur)
B 4.2.	266 Zellen	(6 Spuren)
B 4.3.	400 Zellen	(8 Spuren)
B 4.4.	175 Zellen	(4 Spuren)
B 4.5.	400 Zellen	(8 Spuren)

7.1.5. **Servicebausteine**

In den drei Servicebausteinen sind die folgenden Programme zusammengefaßt:

B 6.1./5	Protokollprogramm, Ansprung mit
B 6.1./8	D6, D7
B 6.2.	Bereichsausschreiben, Ansprung mit D3, D4, D5
B 6.3.	Vergleichsprogramm, Ansprung mit B0

Für den Speicherplatzbedarf (oktal) gilt:

B 6.1./5 und B 6.1./8	534 Zellen	(11 Spuren)
B 6.2.	70 Zellen	(2 Spuren)
B 6.3.	40 Zellen	(1 Spur)

Das Hilfsprogramm, das den Druck des Inhaltes der Zelle 7777 in den verschiedenen Darstellungen mittels D0, D1, D2 ermöglicht, bildet keinen gesonderten Baustein. Es ist in einem Ansprungprogramm für die Servicebausteine untergebracht. Dieses Ansprungprogramm belegt fest die Speicherzellen 400–437. Es ist Voraussetzung für den Ansprung aller mit Hilfe der Kennzeichen „B" und „D" erreichbaren Serviceprogramme (B0, D0/1/2/3/4/5/6/7/8). Werden die Servicebausteine nicht benötigt, so können auch die Speicherplätze 400–437

anderweitig verwendet werden. Von diesem Ansprungprogramm gibt es zwei Ausführungen für die wahlweise Verwendung des 5- und 8-Kanal-Codes.

7.2. Zusammenbau der Bausteine

7.2.1. Abspeicherung der Bausteine

Als einziger Baustein belegt das Minimalprogramm fest die Speicherzellen 0–337 (oktal). Bei Benutzung der Eingabebausteine B 2.1. und B 2.3. befinden sich auf der Spur 340–377 Konstanten (vgl. 7.1.2.). Auf den Zellen 400–437 ist das Programm zum Ansprung der Servicebausteine abgespeichert.

Alle anderen Bausteine können auf beliebige Speicherplätze gebracht werden. Sie sind so aufgebaut, daß sie auch auf sperrbaren Spuren gespeichert werden können. Die Arbeitszellen aller Programme befinden sich bis auf eine Ausnahme auf den Spuren 7600, 7640, 7700 und 7740. Eine Ausnahme bildet der Baustein B 6.1. Das Protokollprogramm benutzt Zellen der Spur 7500–7537 als Arbeitszellen. Auf der Spur 7540–7577 sind die Ansprungzellen für alle nicht im Minimalprogramm enthaltenen Eingabe-, Ausgabe- und Rechenprogramme untergebracht, die als Unterprogramme verwendet werden können.

Sowohl bei Benutzung des Fernschreib-Codes als auch des R 300-Codes muß der Baustein B 1.1./5 bzw. B 1.1./8 – das Minimalprogramm – auf dem Speicher stehen. Wird ein Codewechsel vorgenommen, müssen der Baustein B 1.1. sowie alle anderen Eingabebausteine B 2.1., B 2.2., B 2.3., B 2.4. und B 2.5. ausgetauscht werden. Bei der Eingabe mit Eingabekennzeichen vermittelt das Minimalprogramm den Anschluß der entsprechenden Eingabebausteine B 2.1., B 2.2., B 2.3. und B 2.5. Außerdem benutzen die anderen Programme zum Teil Konstanten und Unterprogramme des Bausteins B 1.1.

Bei allen Programmen des Bausteininterpretiersystems ist besonderer Wert auf Zeitoptimalität gelegt worden. Die Programme arbeiten jedoch nur dann zeitoptimal, wenn die Bausteine ab dem Anfang einer Spur gespeichert sind. Bei allen Bausteinen mit Ausnahme von B 4.1., B 6.2. und B 6.3. ist die Speicherung ab Spuranfang sogar Voraussetzung für die richtige Arbeitsweise der Programme. Bei allen Eingabebausteinen und den Bausteinen B 4.2. und B 4.3. wird darüber hinaus die Speicherung ab einer Zelle mit der Adresse xx00 verlangt.

7.2.2. Anschluß- und Ansprungzellen

Es ist bereits mehrfach darauf hingewiesen worden, daß das Minimalprogramm die Kennzeichen $K_1 = 3, 4, 5, 6, 7, 8, 9, +$, B und D nicht selbst verarbeitet, sondern nur die Anschlußbedingungen für die entsprechenden Lese- bzw. Serviceprogramme enthält. Wird z. B. ein erstes Kennzeichen $K_1 = 5$ eingegeben, so muß außer dem Minimalprogramm noch der Baustein B 2.2. abgespeichert sein. Das Kennzeichen $K_1 = 5$ bewirkt dann einen Sprung zu dem in B 2.2. enthaltenen Eingabeprogramm für Dezimalzahlen. Nach dem Einlesen der Zahl erfolgt die Rückkehr zum Minimalprogramm. Dieses speichert die Zahl ab und geht über zur Eingabe eines neuen ersten Kennzeichens. Für die Herstellung der Anschlußbedingungen im Minimalprogramm ist es notwendig, daß beim Einlesen eines Eingabebausteins B 2.1., B 2.2., B 2.3., B 2.5. bestimmte Zellen im Minimalprogramm versorgt werden. Das heißt, dem Minimalprogramm muß mitgeteilt werden, wohin der jeweilige Baustein abgespeichert wird.

Bei der 5- und bei der 8-Kanal-Variante des Minimalprogramms werden Speicherplätze der Spur 200–237 als Anschlußzellen verwendet.

Den Kennzeichen K_1 entsprechen im einzelnen die folgenden Zellen:

		8-Kanal-Variante	5-Kanal-Variante
$K_1 = 3$	≙	203	201
$K_1 = 4$	≙	204	212
$K_1 = 5$	≙	205	220
$K_1 = 6$	≙	206	225
$K_1 = 7$	≙	207	207
$K_1 = 8$	≙	210	206
$K_1 = 9$	≙	211	230
$K_1 = +$	≙	220	221
$K_1 = B$	≙	222	231
$K_1 = D$	≙	224	211

Bei der Abspeicherung eines Eingabebausteins B 2.1., B 2.2., B 2.3. oder B 2.5. werden die entsprechenden Anschlußzellen automatisch versorgt. Dazu ist es notwendig, daß vor dem Einlesen die Anfangsadresse nicht nur als Hauptleitadresse, sondern auch als 2. Leitadresse gegeben wird. Soll z. B. ein Baustein ab der Zelle 1500 gespeichert werden, so ist vor der Eingabe des Lochbandes

21 1500.

22 1500.

zu geben.

Die Angabe der Anfangsadresse als 2. Leitadresse bewirkt, daß die Anschlußzellen mit einem Sprung zum entsprechenden Leseprogramm gefüllt werden. So stehen z. B. nach dem Einlesen des Bausteins B 2.2./8 in den Zellen 205 und 211 Sprünge zu den Leseprogrammen für eine bzw. für einen Satz von Dezimalzahlen.

Damit die Anschlußzellen im Minimalprogramm gefüllt werden, ist Voraussetzung, daß die Spur 200–237 beim Einlesen eines betreffenden Bausteins nicht gesperrt ist (vgl. Bedienungsanleitung). Auch bei der Eingabe der Serviceprogramme, d. h. der Bausteine B 6.1., B 6.2., B 6.3. und B 4.1. müssen Anschlußzellen versorgt werden. Diese Anschlußzellen befinden sich im Ansprungprogramm für die B- und D-Programme, das die Zellen 400–437 belegt.

Bei Benutzung der Serviceprogramme vermitteln folgende Zellen den Anschluß:

			5- und 8-Kanal-Variante
B 6.2.	D3	≙	403
	D4	≙	404
	D5	≙	405
B 6.1.	D6	≙	406
	D7	≙	407
B 4.1.	D8	≙	410
B 6.3.	B0	≙	430

Die Kombinationen B1 bis B7 werden im BIS nicht verwendet. Sie können vom Benutzer zum Ansprung häufig benötigter Programme verwendet werden. Dazu ist es notwendig, daß die entsprechenden Zellen 431 bis 437 mit Sprungbefehlen zu den jeweiligen Programmen gefüllt werden.

In beiden Codes werden dieselben Speicherplätze als Anschlußzellen verwendet. Auch diese Zellen werden beim Einlesen der Lochbänder für die Bausteine B 6.1., B 6.2., B 6.3. und B 4.1. automatisch gefüllt, wenn die Anfangsadresse des Programms als Hauptleit- und als 2. Leitadresse gegeben wird. Außerdem muß der Sperrschalter für die Spur 400 geöffnet sein (vgl. Bedienungsanleitung). Bei der Versorgung der Anschlußzellen auf der Spur 7540–7577 wird analog verfahren. Die Angabe der Anfangsadresse des Bausteins als 2. Leitadresse bewirkt auch hier, daß die betreffende Zelle auf der Spur 7540 mit

der Anschlußbedingung (im allgemeinen ist das ein Sprung zum jeweiligen Unterprogramm) gefüllt wird.

In Tabelle 1 sind alle nicht im Baustein B 1.1. enthaltenen Programme des BIS und ihre Ansprung- bzw. Anschlußzellen mit den Anschlußbedingungen zusammengestellt.

7.2.3. Verwendung der Bausteine

Bei der Programmierung mittels des Bausteininterpretiersystems ist zu berücksichtigen, daß in allen Bausteinen markierte Einzelbefehle benutzt werden, d. h. durch die Verwendung eines Unterprogramms des Interpretiersystems wird ein bestehender Vergleichszustand verändert. Demzufolge können Vergleicher nur zwischen dem Aufruf von Unterprogrammen des BIS erzeugt und abgefragt werden.

7.3. Bedienungshinweise

7.3.1. Programmeingabe

Das im Minimalprogramm enthaltene Eingabeprogramm für die Eingabe mit Kennzeichen kann auf der Zelle 0000 mit Hilfe der Eingriffstasten

(a) Schalter „H–R" H drücken

(b) Taste G

(c) Schalter „H–R" R drücken

angesprungen werden.

Nach dem Drücken von HGR wartet der Automat auf die Eingabe eines ersten Kennzeichens K_1.
Die Eingabe eines Programms soll an einem Beispiel dargestellt werden:

In einem Hauptprogramm werden zwei Unterprogramme verwendet, die mit 1. UP und 2. UP bezeichnet sind. Das 1. UP benutzt seinerseits das 2. UP als Unterprogramm UUP. Das Hauptprogramm HP soll ab der Zelle 3000, das 1. UP ab der Zelle 3400 und das 2. UP ab der Zelle 3500 gespeichert werden. Das Hauptprogramm ist relativ auf dem Speicherplatz 0004 anzuspringen.

Nach dem Ansprung des Eingabeprogramms durch HGR sind die Hilfsadressen, falls sie nicht auf dem Lochband stehen, folgendermaßen einzugeben:

(Schalter „T–S" auf „T"):

(a) Eingabe des 2. UP: Eingabe der HLA: 21 3500 .
Einlesen des Bandes: H K1 R bei Leser 1
H K2 R bei Leser 2

(b) Eingabe des 1. UP:
Anwahl der Eingabetasten
Eingabe der HLA: 21 3400 .
Eingabe der 2. LA: 22 3500 .
Einlesen des Bandes

(c) Eingabe des Hauptprogramms HP:
Anwahl der Eingabetasten
Eingabe der HLA: 21 3000 .
Eingabe der 2. LA: 22 3400 .
Eingabe der 3. LA: 23 3500 .
Einlesen des Bandes

(d) Ansprung des Hauptprogramms:
Falls die HLA 3000 nicht zerstört ist kann das Programm durch einen Sofortbefehl
„11 4 000."

angesprungen werden. Ist die HLA 3000 nicht mehr erhalten, muß ein Sprungbefehl ohne Adressenänderung

„10 3004 000."

gegeben werden.

7.3.2. Druck des Inhaltes einer beliebigen Zelle

(a) Der Inhalt eines Speicherplatzes $a_1 a_2 a_3 a_4$ kann durch einen Sofort-Lesebefehl

„10 $a_1 a_2 a_3 a_4$ 211."

in den Akkumulator und durch anschließendes Drücken von HGÜE nach der Zelle 7777 transportiert werden.

Nachdem erneut das Eingabeprogramm angesprungen wurde, kann der Inhalt von 7777 durch D0, D1, D2 wahlweise in Triadendarstellung, als Gleitkommazahl oder als ganze Dezimalzahl ausgeschrieben werden. Für den Druck in Gleit- bzw. Dezimaldarstellung ist Voraussetzung, daß der Baustein B 4.2. bzw. B 4.4. abgespeichert ist. Insgesamt ergibt sich somit ohne den Lesebefehl die Kombination:

$$\text{H G Ü E G R D} \quad \begin{Bmatrix} 0 \\ 1 \\ 2 \end{Bmatrix}$$

Nach dem Druck wartet der Automat auf die Eingabe des ersten Kennzeichens K_1 (mit Anwahl von Kanal 2).

(b) Der Inhalt der Zelle $a_1 a_2 a_3 a_4$ ($a_1 a_2 a_3 a_4 \neq$ 7645, 7647, 7657, 7660, 7661, 7705, 7706, 7710, 7711, 7713, 7714, 7722, 7723, 7725, 7730, 7731, 7737, 7746, 7750, 7753, 7755, 7764, 7771, 7773) kann auch mit Hilfe des Programms „Bereichsausschreiben" ausgegeben werden. Dazu muß nach der Eingabe der Adresse $a_1 a_2 a_3 a_4$ eine Endadresse $b_1 b_2 b_3 b_4 \leqq a_1 a_2 a_3 a_4$ gegeben werden oder nach Eingabe von $a_1 a_2 a_3 a_4$ braucht lediglich zweimal das Schlußzeichen „Punkt" gegeben zu werden.

Damit ergibt sich die Kombination:

$$\text{(HGR)} \quad \text{D} \quad \begin{Bmatrix} 3 \\ 4 \\ 5 \end{Bmatrix} \quad a_1 a_2 a_3 a_4 \,.\;.$$

Bei Benutzung von D4 und D5 kommen zu den oben angegebenen nicht ausschreibbaren Zellen die Arbeitszellen der Bausteine B 4.2. bzw. B 4.4. hinzu.

7.3.3. Korrektur von fehlerhaften Plätzen

Als Beispiel stehen auf den Plätzen 3010, 3011 und 3127 falsche Informationen. Das zugehörige, relativ geschriebene Programm wurde ab 3000 gespeichert.

Die Korrektur kann mit Hilfe der Relativadresse (RA) durchgeführt werden:

(a) Eingabe der HLA: 21 3000.
(b) Eingabe der Relativadresse (RA): 20 10.
(c) Eingabe der Sollinformation der Zelle 10 laut Programmformular
(d) Eingabe der Sollinformation der Zelle 11 laut Programmformular
(e) Eingabe der Relativadresse (RA): 20 127.
(f) Eingabe der Sollinformation der Zelle 127 laut Programmformular

7.3.4. Druck des Befehlszählers und Akkumulators

Nach Programmunterbrechungen (Überlauf, Stopp usw.) ist es von Vorteil, den Befehlszählerstand und den Akkumulatorinhalt zu kennen. Da der C 8205 hierfür keine direkte Anzeigemöglichkeit besitzt, ist im Minimalprogramm ein entsprechendes Druckprogramm enthalten.

Nach dem Ausschreiben des Befehlszählerstandes und Akkumulatorinhaltes kann die Abarbeitung des Hauptprogrammes fortgesetzt werden (mit Ausnahme von Unterbrechungen in den Programmen „Adressendruck", „Befehlsdruck" und „Triadendruck").

Im ersten Schritt nach Programmunterbrechungen wird mit Hilfe von HGÜE der Akkumulatorinhalt in die Zelle 7777 transportiert. Durch die Tastenkombination

H G M ↓ E M ↑ „SP Lösen" R

ist anschließend das Druckprogramm anzuspringen (vor dem Drücken von „R" ist der Schalter „SP" zu lösen). Ausgegeben wird die Adresse des zuletzt abgearbeiteten Befehls und der zugehörige Akkumulatorinhalt in Triadendarstellung. Danach wird die Abarbeitung des Programms fortgesetzt, oder der Automat geht in den Zustand „bedingter Stopp", wenn der Schalter „BS" eingeschaltet ist.

War der letzte im Hauptprogramm abgearbeitete Befehl ein Ausgabebefehl, so wird dieser bei Fortsetzung des Programms noch einmal ausgeführt.

7.3.5 Ansprung des Kleinen Eingabeprogrammes

Das Kleine Eingabeprogramm (KEP) kann auf dem Speicherplatz 0002 angesprungen werden. Nach dem Ansprung ist manuell der Eingabekanal anzuwählen. Die Eingabe kann über die Tastatur, die beiden Lochbandleser und die Schreibmaschine erfolgen.

Folgende Tasten sind beim Ansprung des KEP zu drücken:

(a)	Schalter „H-R"	–	H drücken
(b)	Taste G		
(c)	Taste M	–	gedrückt lassen
(d)	Taste E	–	zweimal anschlagen
(e)	Taste M	–	loslassen
(f)	K0, K1, K2	–	anwählen
(g)	Schalter „H-R"	–	R drücken

Danach geht der Automat in den Zustand „Eingabe" und ist zum bitweisen Einlesen über den angewählten Kanal bereit.

7.3.6. Eingabe eines bitweise gelochten Programms

Es soll das Lochband für den Baustein B 1.1. (Minimalprogramm), die Konstanteneingabespur sowie das Ansprungprogramm für die B- und D-Programme, das bitweise gelocht ist, mittels KEP über Leser 1 eingegeben werden. Die angegebenen Programme belegen die Speicherplätze 40–437. Es wird angenommen, daß der Leitbefehl nicht auf dem Lochband steht. (In der Praxis ist der Leitbefehl auf dem Lochband enthalten.)

(a) Ansprung des KEP:

H G M ↓ E E M ↑ (T) R

(b) Aufbau des Abspeicherbefehls „40 240"

(Transport nach Zelle 40) als Leitbefehl:

Entsprechend der internen Darstellung des Befehls

000..... .....000 100 000 010 100 000

ist, beginnend mit dem ersten von Null verschiedenen Bit, die Zeichenfolge:

100 000 010 100 000:

einzugeben.

Durch das Schlußzeichen „:" wird das im Akkumulator aufgebaute Wort zum Leitbefehl erklärt und in einer festen Zelle (7723) gespeichert.

(c) Eingabe über Leser 1:

Nach Eingabe von Doppelpunkt ist – um Kanal 1 anwählen zu können – auf „Halt" zu schalten. Nach dem Drücken von K1 ist der Rechner bereit, das erste Wort über Leser 1 einzulesen. Nach dem Aufbau des Wortes im Akkumulator wird durch das Schlußzeichen Punkt „." der Leitbefehl „40 240" angesprungen. Er bewirkt, daß das Wort nach der Zelle 40 transportiert wird. Durch die anschließende Erhöhung der Adresse des Leitbefehls um eine Einheit wird die Abspeicherung des nächsten Wortes vorbereitet.

Bei der bitweisen Eingabe gibt es kein Lochbandschlußzeichen.

8. Herstellung und Korrektur des Lochbandes

Als Eingabemedium werden beim C 8205 ein Fünf- oder Achtkanal-Lochband verwendet. Die Codierung der Informationen erfolgt im „Internationalen Fernschreib-Code Nr. 2" bzw. im R 300-Code (vgl. Anleitung zur Programmierung in der Maschinensprache). Zur Herstellung des Fünfkanal-Lochbandes kann ein Fernschreiber mit Streifenlocher oder die im Fernschreib-Code ausgelegte elektrische Schreibmaschine „Cellatron SE 5 L" verwendet werden. Das Achtkanal-Lochband kann mit Hilfe einer elektrischen Schreibmaschine „Cellatron SE 5 L" erzeugt werden, die im R 300-Code ausgelegt ist.

Die Befehle, Hilfsadressen usw. werden unmittelbar vom Programmvordruck abgelocht.

Dabei ist bei der
— Eingabe mit Kennzeichen bei dem ersten Kennzeichen
— Eingabe ohne Kennzeichen mit der ersten Ziffer (bzw. dem negativen Vorzeichen) des Wortes zu beginnen.

Zur Gliederung des Lochbandes kann das Zeichen „Zwischenraum (ZR)" benutzt werden. Es muß dann nach dem Schlußzeichen eines Wortes gegeben werden und hat in diesem Zusammenhang weiter keine Funktion.

Für die Eingabe im Fernschreib-Code sind im Zusammenhang mit der Satzeingabe die Umschaltzeichen „Bu" und „Zi/Zei" interessant. Vor der Eingabe des Irrungszeichens „H", das die Löschung eines Satzes bewirkt, muß beim Fernschreiber oder der Cellatron SE 5 L auf „Buchstaben" umgeschaltet werden. Durch das zugehörige Leseprogramm werden die gestanzten Umschaltzeichen übergangen (vgl. Abschnitt 2.2.4.).

Bei der Eingabe mit Kennzeichen werden in beiden Codes alle nicht als erste Kennzeichen definierten Zeichen, d. h. insbesondere die Zeichen „Bu", „Zi/Zei", „Wagenrücklauf", „Zeilenvorschub", wenn sie als erste Kennzeichen auftreten, überlesen und zur Eingabe eines neuen K_1 übergegangen.

Bei Satzeingabe hat das Auftreten der Zeichen „Wagenrücklauf", „Bu", „Zi/Zei" zwischen zwei Zahlen keine Wirkung. Wird das Zeichen „Zeilenvorschub" gelocht, so muß anschließeend „Zwischenraum" gegeben werden (vgl. Abschnitt 2.2.4.).

Eine Korrektur von Falschlochungen kann folgendermaßen vorgenommen werden:

(a) Der Fehler wird bemerkt, bevor das Schlußzeichen des Wortes gelocht ist: Wenn das fehlerhafte Symbol den Eingabebefehl nicht in einen Sprungbefehl umstellt, kann durch Geben von „ZR" der Akkumulatorinhalt gelöscht werden (Sprungumstellende Zeichen, vgl. Anleitung zur Programmierung in der Maschinensprache). Das Wort wird nochmals (bei der Eingabe ohne Kennzeichen beginnend mit der ersten Ziffer oder dem negativen Vorzeichen) abgelocht. Diese Fehlerkorrektur ist auch bei einem falschen ersten oder zweiten Kennzeichen möglich.

Wenn das falschgelochte Zeichen den Befehl „Eingabe-Halt" in einen Sprungbefehl umstellt, so ist dieses zuzulochen. Anschließend ist „ZR" zu geben. Bei dem Fernschreiber und der im Fernschreib-Code bzw. R 300-Code ausgelegten Schreibmaschine Cellatron SE 5 L ist diese Korrektur mit dem Hebel „R" möglich (nach hinten drücken und die Type „→" anschlagen).

Bei der im Fernschreib-Code arbeitenden „Cellatron SE 5 L" wird dabei gleichzeitig auf „Buchstaben" umgeschaltet, so daß danach „Zi" gedrückt werden muß.

Falls die Lochkombination es zuläßt, kann das falsche Zeichen – unter Benutzung des Hebels „R" – mit dem richtigen überlocht werden.

(b) Fehler wird bemerkt, nachdem das Schlußzeichen eines Wortes gelocht ist:

Bei der Eingabe mit Kennzeichen wird mit der Relativadresse ein neuer

Speicherbefehl aufgebaut und das fehlerhafte Wort mit dem richtigen überschrieben.

(c) Fehler wird bemerkt, bevor das Schlußzeichen eines Satzes gelocht ist: Bei der Satzeingabe mit und ohne Kennzeichen wird durch Eingabe eines Irrungszeichens (im R 300-Code: „Irrung Satz", „Irrung Block", „Apostroph"; im Fernschreib-Code: „H") erreicht, daß der Abspeicherbefehl zurückgestellt wird. Das heißt, nach dem Lochen eines Irrungszeichens kann der Satz, beginnend mit dem ersten Wert, nochmals abgelocht werden. Die bereits abgespeicherten falschen Zahlen werden durch die richtigen Werte überschrieben.

Tabelle 1

Ansprungzellen und Anschlußbedingungen für die Programme des Bausteininterpretiersystems

Baustein	Programm	Anschlußzelle 5- und 8-Kanal-Variante	⟨Anschlußzelle⟩ 5- und 8-Kanal-Variante
B 2.1.	Eingabe einer Dezimalzahl (Umwandlung in Gleitkommazahl) mit $K_1 = 3$	201 203	02 4000 0253 000. 02 4000 0253 000.
B 2.1.	Eingabe einer Gleitkommazahl mit $K_1 = 4$	212 204	02 4000 0230 000. 02 4000 0213 000.
B 2.2.	Eingabe einer Dezimalzahl (Umwandlung in Festkommazahl, K. h.) mit $K_1 = 5$	220 205	02 4000 0114 000. 02 4000 0114 000.
B 2.3.	Eingabe einer Dezimalzahl mit maximal 8 Ziffern (Umwandlung in Tetraden) mit $K_1 = 6$	225 206	02 4000 0062 000. 02 4000 0060 000.
B 2.3.	Eingabe einer Dezimalzahl mit maximal 8 Ziffern (Umwandlung in Festkommazahl, K. h.) mit $K_1 = 7$	207 207	02 4000 0046 000, 02 4000 0062 000.
B 2.1.	Eingabe von Dezimalzahlen (Umwandlung in Gleitkommazahlen) ohne Kennzeichen im Satz mit $K_1 = 8$	206 210	02 4000 0267 000. 02 4000 0167 000.
B 2.2.	Eingabe von Dezimalzahlen (Umwandlung in Festkommazahlen, K. h.) ohne Kenzeichen im Satz mit $K_1 = 9$	230 211	02 4000 0102 000. 02 4000 0102 000.
B 2.5.	Eingabe von Externbefehlen mit $K_1 = +$	221 220	02 4000 0121 000. 02 4000 0135 000.
B 6.2.	Bereichsausschreiben mit D3	403	02 10 000.
B 6.2.	Bereichsausschreiben mit D4	404	02 26 000.
B 6.2.	Bereichsausschreiben mit D5	405	02 55 000.
B 6.1.	Protokollprogramm mit D6	406 406	02 000. 02 000.
B 6.1.	Protokollprogramm mit D7	407 407	02 144 000. 02 144 000.
B 4.1.	Bereichsausstanzen mit D8	410 410	02 000. 02 000.
B 6.3.	Vergleichsprogramm mit B0	430	02 000.

Baustein	Programm	Ansprungzelle	⟨Ansprungzelle⟩ 5- und 8-Kanal-Variante
B 5.5.	Festkommadivision ohne Rückspeicherung	7540	02 217 000.
B 5.5.	Festkommadivision mit Rückspeicherung	7541	02 73 000.
B 5.4.	Festkommamultiplikation ohne Rückspeicherung	7542	02 132 000.
B 5.4.	Festkommamultiplikation mit Rückspeicherung	7543	02 70 000.
B 5.3.	Gleitkommadivision ohne Rückspeicherung	7544	02 134 000.
B 5.3.	Gleitkommadivision mit Rückspeicherung	7545	02 134 000.
B 5.2.	Gleitkommamultiplikation ohne Rückspeicherung	7546	02 34 000.
B 5.2.	Gleitkommamultiplikation mit Rückspeicherung	7547	02 34 000.
B 5.1.	Gleitkommasubtraktion ohne Rückspeicherung	7550	02 440 000.
B 5.1.	Gleitkommasubtraktion mit Rückspeicherung	7551	02 56 000.
B 5.1.	Gleitkommaaddition ohne Rückspeicherung	7552	02 140 000.
B 5.1.	Gleitkommaaddition mit Rückspeicherung	7553	02 456 000.
B 4.5.	Ausgabe von Tetraden in Dezimalschreibweise ohne Runden	7554	02 275 000.
B 4.5.	Ausgabe von Festkommazahlen in Dezimalschreibweise ohne Runden (Variante II)	7555	00 27 210.
B 4.5.	Ausgabe von Festkommazahlen in Dezimalschreibweise mit Runden (Variante II)	7556	02 14 000.
B 4.4.	Ausgabe von Festkommazahlen in Dezimalschreibweise (Variante I)	7557	02 12 000.
B 4.3.	Ausgabe einer Gleitkommazahl in Dezimalschreibweise ohne Runden	7560	00 27 210.
B 4.3.	Ausgabe einer Gleitkommazahl in Dezimalschreibweise mit Runden	7561	02 156 000.
B 4.2.	Ausgabe einer Gleitkommazahl in Gleitkommadarstellung	7562	02 100 000.
B 4.1.	Stanzen eines Wortes in Triaden mit $K_l = 0$	7563	02 26 000. 02 26 000.
B 2.4.	Eingabe von Dezimalzahlen mit Kommainformation (Umwandlung in Festkommazahlen, K. h.) im Satz ohne Kennzeichen	7564	02 51 000. 02 45 000.
B 2.2.	Eingabe von Dezimalzahlen (Umwandlung in Festkommazahlen, K. h.) im Satz ohne Kennzeichen	7565	02 43 000.

Baustein	Programm	Ansprungzelle	(Ansprungzelle) 5- und 8-Kanal-Variante
B 2.1.	Eingabe von Dezimalzahlen (Umwandlung in Gleitkommazahlen) im Satz ohne Kennzeichen	7566	02 315 000. 02 267 000.
B 2.1.	Eingabe von Gleitkommazahlen im Satz ohne Kennzeichen	7567	02 316 000. 02 270 000.
B 2.3.	Eingabe einer Dezimalzahl mit maximal 8 Ziffern (Umwandlung in Tetraden) ohne Kennzeichen	7570	02 22 210. 02 14 000.
B 2.3.	Eingabe einer Dezimalzahl mit maximal 8 Ziffern (Umwandlung in Festkommazahl, K. h.) ohne Kennzeichen	7571	02 50 000. 02 53 000.
B 2.4.	Eingabe einer Dezimalzahl mit Kommainformation (Umwandlung in Festkommazahl, K. h.) ohne Kennzeichen	7572	02 122 000. 02 53 000.
B 2.2.	Eingabe einer Dezimalzahl (Umwandlung in Festkommazahl, K. h.) ohne Kennzeichen	7573	02 121 000. 02 121 000.
B 2.1.	Eingabe einer Dezimalzahl (Umwandlung in Gleitkommazahl) ohne Kennzeichen	7574	02 335 000. 02 335 000.
B 2.1.	Eingabe einer Gleitkommazahl ohne Kennzeichen	7575	02 336 000. 02 336 000.

9. Beispiele

9.1. Betrag eines Vektors (UP)

$$|K| = \sqrt{\sum_{i=1}^{n} k_i^2}$$

Das Unterprogramm ist relativ zu adressieren. Vom Hauptprogramm sind bereitzustellen:
in der Zelle 0001 : a°
in der Zelle 0002 : n° — 1° (oktal)

Dabei sind — a° die Adresse der Speicherzelle, ab der die Komponenten k_i gespeichert sind;
— n° die Anzahl der Komponenten des Vektors K in oktalen Adresseneinheiten.

Das Ergebnis soll dem Hauptprogramm im Akkumulator rückübermittelt werden.

				⟨Ac⟩	Bemerkungen
0000	01	3	000.		
0001	0.				Parameterzellen (werden vom HP gefüllt)
0002	0.				
0003	01	23	240.		Sicherung Rücksprung
0004	00	7776	241.	0	$\Sigma := 0;\ i := 0$
0005	01	1	210.		
0006	01	24	240.		$i := i + 1$
0007	00		251.	k_i	Lesebef. f. d. Komp. k_i
0010	00	7773	240.		$\langle R_1 \rangle := k_i$
0011	00	7774	240.		$\langle R_3 \rangle := k_i$
0012	00	7547	025.		$\langle R_1 \rangle := k_i^2$
0013	00	7553	025.		$\langle R_1 \rangle := \Sigma + k_i^2$
0014	00	7776	240.		$\langle R_2 \rangle := \Sigma + k_i^2$
0015	01	24	211.		
0016	01	25	210.		
0017	01	1	220.		
0020	01	2	330.		
0021	01	5	140.		$P\ (i \leqq n)$
0022	02		025.	$\lvert K \rvert$	UP Quadratwurzel
0023	0.				Rücksprung
0024	0.				
0025	00	1	000.		

Durch den Sprung in Zelle 0000 werden die Versorgungszellen für die Parameter übergangen und in Zelle 0003 wird der Rücksprung in die Zelle 0023 gespeichert.

Die Zwischensumme wird bei der Betragsbildung im Register R_2 gespeichert; im Befehl 0004 wird sie zu Beginn gleich Null gesetzt. Durch die Befehle 0005 und 0006 werden der Lesebefehl für die erste Komponente des Vektors K bereitgestellt und ein Zähler zur Zyklenzählung aufgebaut. Im Befehl 0007 wird die erste Komponente k_1 gelesen und durch die beiden folgenden Befehle wird k_1 in die Register R_1 und R_3 transportiert. Durch die Multiplikation mit Rückspeicherung wird k_1^2 gebildet und in R_1 gespeichert. Anschließend erfolgt die Addition (die vorherige Teilsumme steht bereits im R_2). Im Befehl 0014 wird das Ergebnis wieder nach R_2 transportiert. Durch die Befehle 0015, 0016, 0006 wird der Lesebefehl für

die folgende Komponente bereitgestellt und gleichzeitig der Zähler um eine Einheit erhöht. In den Befehlen 0017, 0020 und 0021 wird geprüft, ob bereits alle n-Komponenten des Vektors K gelesen worden sind. Ist das nicht der Fall, so wird der Zyklus fortgesetzt. Ist $i^o > n^o$, so wird im Befehl 0022 zum UP Quadratwurzel gesprungen und aus dem im Register R_1 stehenden Quadrat des Betrages wird die Wurzel gezogen. Nach dem Rücksprung aus dem UP Quadratwurzel steht diese im Akkumulator. Anschließend erfolgt im Befehl 0023 der Rücksprung zum Hauptprogramm.

Bei Verwendung dieses Unterprogramms müssen die Bausteine:

B 5.1. Gleitkommaaddition und -subtraktion,

B 5.2. Gleitkommamultiplikation

abgespeichert sein. Außerdem muß das UP Quadratwurzel auf dem Speicher stehen.

9.2. Matrizenmultiplikation

Es soll das Matrizenprodukt C = A · B berechnet werden. Die Matrizen A, B, C haben die Gestalt:

$$A = (a_{ij}) \quad \begin{matrix} (i = 1, \ldots, m) \\ (j = 1, \ldots, n_1) \end{matrix}$$

$$B = (b_{ij}) \quad \begin{matrix} (i = 1, \ldots, n_1) \\ (j = 1, \ldots, n_2) \end{matrix}$$

$$C = (c_{ij}) \quad \begin{matrix} (i = 1, \ldots, m) \\ (j = 1, \ldots, n_2) \end{matrix}$$

Die Elemente c_{kl} der Ergebnismatrix C werden nach der Formel

$$c_{kl} = \sum_{j=1}^{n_1} a_{kj} \, b_{jl}$$

berechnet.

Es wird vorausgesetzt, daß die Elemente b_{ij} der Matrix B in der Reihenfolge $b_{11} b_{12} \ldots b_{1 n_2} b_{21} b_{22} \ldots b_{2 n_2} \ldots b_{n_1 1} b_{n_1 2} \ldots b_{n_1 n_2}$ als Gleitkommazahlen im Rechner gespeichert sind und daß die Elemente a_{ij} von A als Dezimalzahlen in der Form

$a_{11} a_{12} a_{13} \ldots\ldots a_{1 n_1}$ SM

$a_{21} a_{22} a_{23} \ldots\ldots a_{2 n_1}$ SM

. . . .

$a_{m1} a_{m2} a_{m3} \ldots\ldots a_{m n_1}$ SM BM

abgelocht sind. Das heißt, nach jeder Zeile von A ist das Zeichen „Satzmarke" gelocht, und auf die letzte Satzmarke folgt das Zeichen „Blockmarke".

Vom Programm wird über den Leser 2 immer nur eine Zeile a_{kj} (j = 1, ..., n_1) der Matrix A eingelesen. Nach der Berechnung der Elemente c_{kl} (l = 1, ..., n_2) werden diese in Dezimalschreibweise über die Schreibmaschine und in Triadendarstellung über den Lochbandstanzer ausgegeben.

Das Druckbild hat die Gestalt:

$c_{11} c_{12} \ldots\ldots c_{1 n_2}$

$c_{21} c_{22} \ldots\ldots c_{2 n_2}$

. . . .

$c_{m1} \; c_{m2} \ldots\ldots c_{m n_2}$

Das Programm ist als Unterprogramm aufgebaut und relativ adressiert. Vom Hauptporgramm sind bereitzustellen:

in der Zelle 0001 : $n_1^0 - 1^0$

in der Zelle 0002 : n_2^0 (oktal)

in der Zelle 0003 : b^0

Dabei sind n_1^0 — die Anzahl der Spalten von A (= Anzahl der Zeilen von B) in oktalen Adresseneinheiten

n_2^0 — die Anzahl der Spalten von B in oktalen Adresseneinheiten

b^0 — die Adresse der Zelle, ab der die Elemente von B gespeichert sind.

		⟨Ac⟩	Bemerkungen
0000	01 4 000.		
0001	0.		Parameterzellen (werden vom HP
0002	0.		gefüllt)
0003	0.		
0004	01 13 240.		Sicherung RS, k := O
0005	00 4500 030.		AW K2
0006	00 2720 030.		Dr (WRZL)
0007	01 54 211.	01 100 240.	k := k + 1
0010	00 7745 240.		Aufbau d. Abspeicherbef.
0011	00 7566 025.		E (a_{kj}, j = 1, . . . , n_1; SM, BM)
0012	01 14 001.		Spr. b. Eing. v. „SM"; l := 1
0013	0.		Rückspr. b. Eing. v. „BM"
0014	01 3 210.		
0015	01 56 240.		
0016	01 2 220.		
0017	01 57 240.		
0020	00 7776 241.	0	C_{kl} := 0 j := 1
0021	01 54 210.	r + 01 100 240.	
0022	01 60 240.		
0023	00 251.	a_{kj}	
0024	00 7774 240.		⟨R_3⟩ := a_{kj}
0025	01 57 211.		
0026	01 2 210.		
0027	01 57 240.		
0030	00 251.	b_{jl}	
0031	00 7773 240.		⟨R_1⟩ := b_{jl}
0032	00 7547 025.		⟨R_1⟩ := $a_{kj} \cdot b_{lj}$
0033	00 7553 025.		⟨R_1⟩ := $c_{kl} := c_{kl} + a_{kj} \cdot b_{jl}$
0034	00 7776 240.		⟨R_2⟩ := c_{kl}
0035	01 60 211.		
0036	01 55 210.	r + 00 1 000.	j := j + 1
0037	01 54 220.	r — 01 100 240.	
0040	01 1 330.		
0041	01 21 140.		P (j ≦ n_1)
0042	00 4500 030.		AW K2
0043	00 222 7561 025.		Dr (c_{kl})
0044	00 4600 030.		AW K1
0045	00 7563 025.		St (c_{kl})
0046	01 56 211.		
0047	01 55 210.		l := l + 1
0050	01 3 220.		
0051	01 2 330.		
0052	01 5 150.		P (l ≦ n_2)
0053	01 14 000.		
0054	01 100 240.		Abspeicherbefehl
0055	00 1 000.		
0056	0.		
0057	0.		Arbeitszellen
0060	0.		

Der Sprung in Zelle 0000 übergeht die Versorgungszellen für die Parameter und führt in Zelle 0004 zur Speicherung des Rücksprunges in die Zelle 0013. Nach der Anwahl von Kanal 2 (Leser 2 und Schreibmaschinenausgabe) wird ein Wagenrücklauf mit Zeilenschaltung ausgeführt.

Durch die Befehle 0007 und 0010 wird der Abspeicherbefehl für die Elemente a_{kj} ($j = 1, \ldots, n_1$) einer Zeile der Matrix A bereitgestellt. Im Befehl 0011 wird zum UP „Eingabe von Zahlen in Dezimalschreibweise im Satz ohne KZ" gesprungen und eine durch das Zeichen „Satzmarke" abgeschlossene Zeile von A eingelesen. Die Zahlen werden intern in Gleitkommazahlen umgewandelt und ab der Zelle 0100 gespeichert. Das Zeichen „Satzmarke" bewirkt die Rückkehr zur Zelle 0012, und von dort wird zur Zelle 0014 gesprungen. Gleichzeitig wird der Akkumulatorinhalt gelöscht. In den Befehlen 0014 bis 0017 sowie 0025 bis 0027 wird die Adresse des Befehls für das spaltenweise Lesen der Elemente b_{jl} ($j = 1, \ldots, n_1$) aufgebaut. Die Zwischensumme bei der Berechnung eines Elementes c_{kl} wird in R_2 gespeichert. In Zelle 0020 wird diese Zwischensumme zu Beginn gleich Null gesetzt. Durch die Befehle 0021 und 0022 wird die Adresse des Lesebefehls für die Elemente a_{kj} ($j = 1, \ldots, n_1$) bereitgestellt, und im Befehl 0023 wird das Element a_{kj} gelesen.

Durch den Befehl 0024 wird a_{kj} nach R_3 transportiert. In den Befehlen 0030 und 0031 erfolgt das Lesen und der Transport von b_{jl} nach R_1. Im Befehl 0032 wird das Produkt von $a_{kj} \cdot b_{jl}$ gebildet und nach R_1 gespeichert. Im Befehl 0033 wird das Produkt $a_{kj} \cdot b_{jl}$ zu der in R_2 stehenden Zwischensumme addiert und nach R_1 und R_2 transportiert.

Durch die Befehle 0035 bis 0041 wird die Adresse des Lesebefehls für die Elemente a_{kj} um eine Einheit erhöht, und anschließend wird geprüft, ob bereits alle Elemente a_{kj} ($j = 1, \ldots, n_1$) gelesen worden sind. Ist das nicht der Fall, so werden das nächste a_{kj} und b gelesen, multipliziert, und das Produkt wird zur Zwischensumme in R_2 addiert.

Nach der Berechnung des Elementes c_{kl} wird im Befehl 0043 zum UP „Druck einer Gleitkommazahl in Dezimalschreibweise mit Runden" gesprungen und das in R_1 stehende c_{kl} als Dezimalzahl über die Schreibmaschine ausgegeben. Der Druck erfolgt mit 9 Stellen vor und 2 Stellen nach dem Komma.

Im Befehl 0045 wird das c_{kl}, das noch im Register R_1 gespeichert ist, in Triadendarstellung über den Lochbandstanzer ausgegeben. In den Befehlen 0046 bis 0053 wird die Adresse des Befehls für das Lesen der nächsten Spalte der Matrix B bereitgestellt, und es wird geprüft, ob alle Spalten von B mit der Zeile a_{kj} ($j = 1, \ldots, n_1$) multipliziert worden sind. Ist das der Fall, d. h. alle Elemente c_{kl} ($l = 1, \ldots, n_2$) einer Zeile der Matrix C sind berechnet worden, so wird im Befehl 0006 ein Wagenrücklauf mit Zeilenschaltung ausgeführt, und der Zyklus beginnt von neuem. Es wird die folgende Zeile von A eingelesen und damit eine weitere Zeile von C bestimmt.

Nach dem Einlesen aller m-Zeilen von A und der Berechnung von m-Zeilen von C wird bei der Eingabe des Zeichens „Blockmarke" im Befehl 0011 zur Zelle 0013 gesprungen. Von dort erfolgt der Rücksprung zum Hauptprogramm.

Bei Abarbeitung des Programms werden die Bausteine B 2.1./8, B 5.1., B 5.2., B 4.1./5 oder B 4.1./8 und B 4.3. benötigt.

9.3. Beispiel für die Anwendung der Festkommaoperationen und der Externbefehle

Es soll eine Zeitfondsbilanzierung durchgeführt werden. Für Kostenstellen K_i wird der geplante Zeitfonds T_f^i und der benötigte Zeitfonds T^i gegenübergestellt. Die Berechnung des benötigten Zeitfonds für eine Kostenstelle K_j erfolgt nach der Formel:

$$T^i = (n \cdot t_S + t_A)^i$$

Dabei sind: n^i – Stückzahl

t_S^i – Stückzeit (in min)

t_A^i – Vorbereitungs-, Einrichtezeit (in min)

Außerdem soll die prozentuale Auslastung

$$T_p^i = \frac{100\,T^i}{T_f^i}$$

ermittelt werden.

Die Daten sind in der Reihenfolge

$K_i\ T_f^i\ n^i\ t_S^i\ t_A^i$ SM $n^i\ t_S^i\ t_A^i$ SM...... $n^i\ t_S^i\ t_A^i$ SM BM

$K_{i+1}\ T_f^{i+1}\ n^{i+1}\ t_S^{i+1}\ t_A^{i+1}$ SM$n^{i+1}\ t_S^{i+1}\ t_A^{i+1}$ Sm...$n^{i+1}\ t_S^{i+1}\ t_A^{i+1}$ SM BM

....

$K_{i+1}\ T_f^{i+1}\ n^{i+1}\ t_S^{i+1}\ t_A^{i+1}$ SM$n^{i+1}\ t_S^{i+1}\ t_A^{i+1}$ SM... $n^{i+1}\ t_S^{i+1}\ t_A^{i+1}$ SM BM 0

abgelocht.

Das heißt, nach den drei Werten: Stückzahl, Stückzeit und Vorbereitungszeit ist jeweils das Schlußzeichen „Satzmarke" gelocht, und das Kostenstellenende wird durch Ablochen von „Blockmarke" gekennzeichnet. Die letzte Kostenstellennummer „0" wird als Schlußzeichen für die Bilanzierung verwendet.

Das Programm setzt voraus, daß der Datenstreifen über Leser 2 eingelesen wird.

Die Ergebnisse werden in der folgenden Form über die Schreibmaschine ausgegeben:

Kostenstelle	geplanter Zeitfonds	benötigter Zeitfonds	Auslastung in Prozent
K_i	T_f^i	T^i	T_p^i

			⟨Ac⟩	Bemerkungen
0000	01	40 240.		Sicherung des Rücksprungs
0001	+	72.		AW K2
0002	+	73.		Dr (WRZL)
0003	+	61.	K^i	E (K_i)
0004	00	7773 340.		$\langle R_1\rangle := K_i$
0005	01	40 170.		P $(K_i = 0)$
0006	+	45 340.		Dr (K_i)
0007	+	61.	T_f^i	E (T_f^i)
0010	01	41 240.		$a := T_f^i$
0011	+	75.		Dr (Tab.)
0012	+	01.		$\langle R_1\rangle := T_f^i$
0013	+	45 450.		Dr (T_f^i)
0014	00	7776 241.	0	$\langle R_2\rangle := \Sigma := 0$
0015	01	12 211.	00 7773 240.	
0016	+	04.		Aufbau des Abspeicherbefehls
0017	+	65.		E $(n^i,\ t_S^i,\ t_A^i$, SM, BM)
				$\langle R_1\rangle := n^i,\ \langle R_3\rangle := t_S^i$
0020	01	33 000.		Sprung b. Eingabe „SM"
0021	+	75.		Dr (Tab.)
0022	00	7776 211.		
0023	+	01.		$\langle R_1\rangle := T^i := \Sigma$
0024	+	45 890.		Dr (T^i)

			Ac	Bemerkungen
0025	01	41 211.		
0026	+	03.		$\langle R_3 \rangle := T_f^i$
0027	+	265.		$\langle R_1 \rangle := T_p^i := \frac{T^i}{T_f^i} \cdot 100$
0030	+	75.		Dr (Tab.)
0031	+	46 642.		Dr (T_p^i)
0032	01	2 000.		
0033	+	15.	$n^i \cdot t_S^i$	
0034	00	7775 210.	$n^i \cdot t_S^i + t_A^i$	
0035	00	7776 210.		
0036	+	02.		$\Sigma := \Sigma + n^i \cdot t_S^i + t_A^i$
0037	01	15 000.		
0040	0.			Rücksprung
0041	0.			Arbeitszelle

Im Befehl 0000 wird der Rücksprung in Zelle 0040 gespeichert. Nach der Anwahl von K2 wird in Befehl 0003 über Leser 2 die erste auf dem Datenstreifen abgelochte Kostenstellennummer eingelesen und in die Zelle 7773 transportiert. Bei dem Transportbefehl in Zelle 0004 wird gleichzeitig ein Vergleicher gesetzt. Ist die letzte Kostenstellennummer 0 eingelesen worden, so wird der Vergleicher 3 erzeugt und durch den Befehl in Zelle 0005 zur Zelle 0040 gesprungen. Von dort erfolgt dann der Rücksprung zum Hauptprogramm. Wird eine Kostenstellennummer ungleich Null eingelesen, so wird der Sprungbefehl in Zelle 0005 nicht ausgeführt, sondern es erfolgt im Befehl 0006 der Druck der Kostenstellennummer als ganze, vierstellige Dezimalzahl.

Anschließend wird im Befehl 0007 der für die Kostenstelle K_i geplante Zeitfonds T_f^i eingegeben, in Zelle 0041 zwischengespeichert und im Befehl 0013 als ganze, fünfstellige Dezimalzahl ausgeschrieben.

Die bei der Berechnung des benötigten Zeitfonds auftretende Zwischensumme wird in Zelle 7776 gespeichert. Sie wird zu Beginn im Befehl 0014 gleich Null gesetzt. Durch die Befehle 0015 und 0016 wird der Abspeicherbefehl für die im Satz einzugebenden drei Werte n^i, t_s^i, t_A^i aufgebaut, und im Befehl 0017 werden die Zahlen eingelesen und in den Zellen 7773, 7774 und 7775 gespeichert. Bei Eingabe des Schlußzeichens „SM" wird zur Zelle 0020 zurückgekehrt und von dort wird zur Zelle 0033 gesprungen. Im Befehl 0033 werden die Inhalte der Register R_1 und R_3 miteinander multipliziert. Das heißt, es wird das Produkt $n^i \cdot t_s^i$ gebildet. Der Korrekturfaktor p_1 ist gleich Null, da n^i und t_s^i ganze Zahlen sind und auch das Produkt ganzzahlig sein soll.

Nach der Multiplikation wird zu dem im Akkumulator stehenden Produkt der in der Zelle 7775 zwischengespeicherte Wert t_A^i addiert. Das ist möglich, da diese Zelle bei den benutzten Bausteinen nicht als Arbeitszelle verwendet wird.

In den Befehlen 0035 und 0036 werden der Akkumulatorinhalt und der Inhalt von R_2 addiert und nach R_2 transportiert. Danach werden weitere Daten eingelesen.

Falls das Satzschlußzeichen „Blockmarke" eingegeben wird, erfolgt die Rückkehr zur Zelle 0021.

Im Befehl 0024 wird zunächst der für die Kostenstelle K_i berechnete Zeitfonds als ganze, neunstellige Dezimalzahl ausgegeben.

Im Befehl 0027 wird die prozentuale Auslastung berechnet. Es wird gebildet

$$\frac{T^i}{T^i_f} \cdot 10^5$$

Der Korrekturfaktor $p_1 = 5$ ergibt sich daraus, daß der Quotient T^i/T^i_f mit 10^2 zu multiplizieren ist und das Ergebnis T^i_p drei Kommastellen besitzen soll. Im Befehl 0031 wird der Wert T^i_p als Dezimalzahl mit vier Stellen vor dem Komma und zwei Stellen nach dem Komma ausgegeben ($p_3 = 6$, $p_2 = 4$, $p_1 = 2$).

(Das gleiche Resultat erhält man, wenn im Befehl 0027 der Korrekturfaktor $p_1 = 4$ und im Befehl 0031 die Parameter $p_3 = 5$, $p_2 = 4$, $p_1 = 2$ angegeben werden.)

Danach wird zur Eingabe einer neuen Kostenstellennummer übergegangen.

Bei der Eingabe des Programms wird der Baustein B 2.5./8 benötigt, während bei der Abarbeitung die Bausteine B 2.2./8, B 4.5., B 5.4. und B 5.5. auf dem Speicher stehen müssen.

9.4. Beispiel für das Druckbild des Protokollprogramms

Ein auf den Speicherplätzen 3000 bis 3016 stehendes Programm soll protokolliert werden.

Dieses Programm multipliziert zwei Gleitkommazahlen, und mit Hilfe des ab der Zelle 3040 gespeicherten Unterprogramms „Quadratwurzel" wird die Wurzel des im Register R_1 stehenden Produktes bestimmt. Nach der Anwahl von Kanal 2 wird das Ergebnis in Gleitkommaschreibweise ausgegeben. Danach erfolgt ein unbedingter Stopp.

```
3000 00 3002 000
3001  4  9000000 01
3002 00 3001 211
3003 00 7773 240
3004 00 3006 000
3005  4  4000000 01
3006 00 3005 211
3007 00 7774 240
3010 00 7547 025
3011 00 3040 025
3012 00 7773 240
3013 00 4500 030
3014 00 2720 030
3015 00 7562 025
3016 00 0000 020
```

Nach dem Ansprung des Protokollprogramms durch D6 wird das Kennzeichen „1" eingegeben, da kein Unterprogramm mitprotokolliert werden soll. Anschließend wird die Adresse 3000 des ersten zu protokollierenden Befehls eingegeben. Die folgende Eingabe der Adressen 3002 und 3015 legt fest, daß der Akkumulatorinhalt zwischen den angegebenen Adressen auszuschreiben ist.

Das Eingabebild hat somit die Form:

D6 1 3000. 3002. 3015.

Nach der Eingabe der Adresse 3015 beginnt die Protokollierung. Es ergibt sich das folgende Druckbild:

```
  9000000 01
  9000000 01
    3006
  9000000 01
  4000000 01
  4000000 01
UP 7547
  3600000 02
UP 3040
  6000000 01
  6000000 01
AW 2
  6000000 01

  6000000 01
UP 7562
  6000000 01
—2147484 10
```

Die beiden ersten Zeilen ergeben sich durch die Befehle 3002 und 3003. Danach wird die Zieladresse des unbedingten Sprunges auf 3004 und der Akkumulatorinhalt nach Ausführung des Sprunges ausgegeben. Die darauffolgenden zwei Zeilen sind den Befehlen 3006 und 3007 zugeordnet. Der Aufruf des Standardprogramms „Gleitkommamultiplikation mit Speicherung" wird durch Druck von „UP" und der Ansprungadresse gekennzeichnet. Die nächste Zeile gibt den Inhalt des Akkumulators vor dem Rücksprung ins Hauptprogramm an.

Die beiden folgenden Zeilen ergeben sich in analoger Weise durch die Abarbeitung des Unterprogramms „Quadratwurzel". Die auf dem Speicherplatz 3013 stehende Anwahl von K2 (Leser 2 und Schreibmaschinenausgabe) wird durch den Druck von AW 2 hervorgehoben. Die anschließende Leerzeile entsteht durch die Ausführung des Befehls „Wagenrücklauf mit Zeilenvorschub" in Zelle 3014. Durch „UP 7562" ist der Einsprung in das Druckprogramm für Gleitkommazahlen charakterisiert. Die nächste Zeile ergibt sich durch die Abarbeitung dieses Druckprogramms. Die letzte Zeile stellt den unwesentlichen Inhalt des Akkumulators vor dem Rücksprung aus dem Druckprogramm dar.

Bei der Protokollierung des angegebenen Beispiels werden die Bausteine B 4.2., B 5.2. und B 6.1. benötigt. Außerdem muß das UP Quadratwurzel auf dem Speicher stehen.

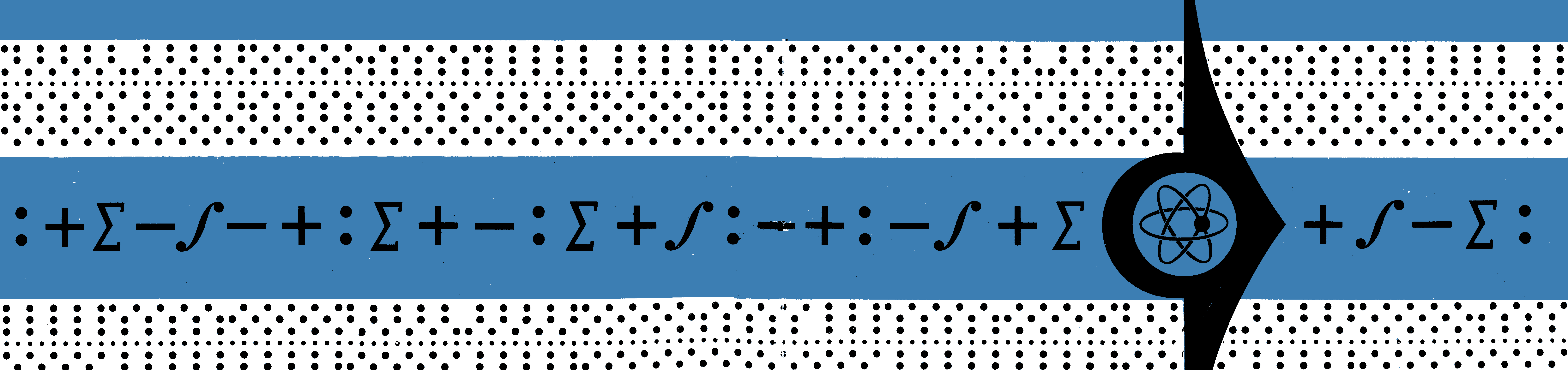


VEB
KOMBINAT ZENTRONIK
RECHENELEKTRONIK
MEININGEN/ZELLA MEHLIS

PROGRAMMIERUNG UND BEDIENUNG DES DIGITALEN KLEINRECHNERS C 8205

HEFT 4

BAUSTEIN-INTERPRETIERSYSTEM



VEB KOMBINAT ZENTRONIK
Rechenelektronik Meiningen, Sitz Zella-Mehlis
DDR - 606 Zella-Mehlis
Deutsche Demokratische Republik

Exporteur:
BÜROMASCHINEN-EXPORT GmbH.
DDR - 108 Berlin 8
Friedrichstraße 61

9. Seite 50, Punkt 9.1. muß ergänzt werden:

12. Druck des <7777>
(8.2)

0. Seite 51, Punkt 9.1. muß richtig lauten:

Speicherplatzbedarf: 0400_8 ... 0437_8 Ansprung-Programm für Hilfs- und Service-programme

1. Seite 53, Punkt 9.8. ändert sich

(6.4) in (6.5)

2. Seite 54, Punkt 9.9. ändert sich

(6.5) in (6.6)

13. Seite 54, Punkt 9.10. ändert sich

(6.6) in (6.7)

14. Seite 56, Punkt 9.18. entfällt

1. Druck des <7777>

lautet:

9.18. B 6.2

Bereichsausschreiben
(8.3)

Speicherplatzbedarf: 0000_8 ... 0067_8 (relativ)

15. Seite 58, Punkt 10.1., 9. Zeile muß richtig lauten:

... Serviceprogramme (0400_8 ... 0437_8) eingegeben.

16. Seite 59, Punkt 10.2., 2. Satz oben muß richtig lauten:

Bei Eingabe der Bausteine 6.1, 6.2, 4.1 und 6.3 ist die Spur 400 zu öffnen.

Korrekturblatt für C 8205 - Bausteininterpretationssystem

1. Seite 7, Punkt 5 muß richtig lauten:

   Wörter, die mit den KZ "D" und "B" in den Rechner ...

2. Seite 18, Abb. 1 müssen die Adressen in der 4. und 5. Spalte von links

von	7721	in	7621
von	(7727)	in	(7627)
von	7722 ... 7731	in	7622 ... 7631
von	(7723 ... 7730)	in	(7623 ... 7630)

   geändert werden.

3. Seite 22, Punkt 4.2. muß richtig lauten:

   Eingabebild: xxxx $a_1a_2a_3a_4$.

4. Seite 33, Punkt 6.6. muß richtig lauten:

Kommainformation			Druckbild
d	v	n	
15	15	15	ZRZR 1234567800000,000000000000000

5. Seite 37, 10. Zeile von unten:

   ... mit $10^{|q|}$ zu multiplizieren ...

6. Seite 38, 5. Zeile, 14. Zeile
   Seite 39, 4. Zeile, 6. Zeile muß richtig stehen:

   Dividend

   Seite 38, 2. Zeile von unten:

   ... durch $10^{|q|}$ dividiert werden ...

7. Seite 44, Punkt 8.2. muß richtig lauten:

   8.2. Druck des <7777> (B 1.1)

8. Seite 48, Punkt 8.5. muß richtig lauten:

   Ansprung: Sofortbefehl 10 $a_1a_2a_3a_4$ 000.
   oder BO

17. Tabelle 1, 9. und 14. Zeile muß richtig lauten:

2 Hilfsadresse 2 K_2 xxxx $a_1a_2a_3a_4$.
.
.
.
D Ansprung von Hilfs- D
B und Serviceprogrammen B

18. Tabelle 5, 2. Zeile oben ändert sich:
Vorsorgungszelle in Versorgungszelle

19. Tabelle 7 ändert sich:

+15. in +15q.
+25. in +25q.
+16. in +16q.
+26. in +26q.

20. Tabelle 9, 4. und 5. Zeile von unten muß richtig lauten:

10 $a_1a_2a_3a_4$ 000. K $\frac{K\ 1}{K\ 2}$ R   <u>Vergleichsprogramm</u>
B 0 B $\frac{K\ 1}{K\ 2}$ R   K 1 Leser 1
K 2 Leser 2

21. Tabelle 11 ändert sich:

D	224	211	00 4000 0411 000.	00 4000 0411 000.

22. Tabelle 11 wird ergänzt:

B	222	231	00 4000 0424 000.	00 4000 0424 000.
B–0	430		01	0 000.